AD-A043 302   RCA GOVERNMENT SYSTEMS DIV MOORESTOWN N J MISSILE AND--ETC   F/G 17/9
PROGRAMMABLE FFT LINEAR FM WAVEFORM PROCESSOR. PHASES II AND II--ETC(U)
JUL 77   L W MARTINSON, J A LUNSFORD   F33615-74-C-1077

UNCLASSIFIED   AFAL-TR-77-23   NL

1 OF 3
AD
A043302

AFAL-TR-77-23

12

NW

PROGRAMMABLE FFT LINEAR FM WAVEFORM PROCESSOR PHASES II AND III

RCA/GOVERNMENT ~~AND COMMERCIAL~~ SYSTEMS Div
MISSILE AND SURFACE RADAR DIVISION
MOORESTOWN, NEW JERSEY 08057

JULY 1977

TECHNICAL REPORT AFAL-TR-77-23
FINAL REPORT FOR PERIOD FEBRUARY 1974 - NOVEMBER 1976

Approved for public release; distribution unlimited

AIR FORCE AVIONICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any wrights or permission to manufacture, use, sell any patented invention that may in any way be related thereto.

This report has been reviewed by the information office ASD/OIP and is releasable to the National Technical Information Service (NTIS). At NTIS it will be available to the general public, including foreign nations. This technical report has been reviewed and is approved for publication.

Charles T Brodnax
C. T. Brodnax, PhD
Project Engineer
Microelectronics Group
Advanced Electronic Devices Branch
Electronic Technology Division

FOR THE COMMANDER:

William J Edwards
W. J. Edwards
Chief, Electronics Technology Division
Air Force Avionics Laboratory

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19 **REPORT DOCUMENTATION PAGE**		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. ~~REPORT NUMBER~~ 18 AFAL-TR-77-23	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER rept.
4. TITLE (and Subtitle) 6 PROGRAMMABLE FFT LINEAR FM WAVEFORM PROCESSOR, PHASES II AND III,		9 5. ~~TYPE OF REPORT & PERIOD COVERED~~ Final ~~-~~ Feb~~ruary~~ 1974 ~~to~~ Nove~~mbe~~r 1976, 6. ~~PERFORMING ORG.~~ REPORT NUMBER
7. ~~AUTHOR(s)~~ 10 Lloyd W. Martinson John A. Lunsford		8. CONTRACT OR GRANT NUMBER(s) 15 F33615-74-C-1077 new
9. PERFORMING ORGANIZATION NAME AND ADDRESS RCA/Government and Commercial Systems Missile and Surface Radar Division Moorestown, New Jersey 08057		16 10. PROGRAM ELEMENT, PROJECT, TASK ~~AREA & WORK UNIT~~ NUMBERS Project 69CK 17 Task 02
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Avionics Laboratory Air Force Systems Command Wright-Patterson AFB, Ohio		11 12. ~~REPORT DATE~~ July 1977 13. ~~NUMBER OF PAGES~~ 230
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) 12 229p.		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release, distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Signal Processing, Fast Fourier Transforms, Floating Point Arithmetic, Silicon on Sapphire devices, Pipeline Processor, Computers, Simulation, Synthetic Aperture, Pulse Compression, Waveform Generation, Step Transform, Time Sidelobes, Radar, Linear F.M., Digital Processing, Subarrays, Hybrid Packaging, Programmable Processors, Motion Compensation, Large Scale

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the detailed design of a high speed programmable linear FM waveform processor together with the fabrication and testing of the radix-2 pipeline FFTs which are the computational heart of the system. The system features three basic new developments; the step transform algorithm for linear FM and synthetic aperture radar processing, floating point pipeline FFTs and CMOS/SOS LSI circuitry. All of these features serve to increase the processing capability of digital processing systems while reducing system complexity, cost

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

405 296

1B

19. Integration, Partitioning, Modularity, High Speed Logic, Data Reordering, System Testing, Functional Module Testing.

20. and size. Six custom LSI CMOS/SOS circuits were developed for the ssstem including a 9 x 9 multiplier, 9 bit adder and a dual 8 bit scaler for floating point applications. The LSI circuits have been assembled on six functional modules including a complex multiplier, complex adder/subtracter and a programmable FFT memory. The 1.7 inch by 5.6 inch modules were designed using a new packaging concept developed for LSI systems. Circuits are placed in leadless carriers which are assembled on the thick film ceramic modules by reflow soldering. The circuits and modules developed for the system have wide applications in addition to the specific use in the system designed. The modules and pipeline FFTs were tested by interfacing with a PDP-11/20 computer through a test bed which permitted operation of the hardware at the design goal clock speed of 10 MHz. The FFT size is programmable to lengths of 16, 32 and 64 points. A computer simulation of the hardware which exactly duplicates the bit-level logic in the functional modules was used for system test and debugging. The FFT was operated with no errors up to speeds in excess of 5 MHz. The speed was limited by a few circuits which were much slower than expected, a problem which can be avoided in the future by accepting a small reduction in yield to eliminate the low speed circuits.

FOREWORD

This report was prepared by RCA, Government and Commercial Systems, Missile and Surface Radar Division, Moorestown, New Jersey, under USAF Contract F33615-74-C-1077, Project 69CK, Task 02, in response to contract item 0002. This work was administered under the direction of the Air Force Avionics Laboratory, Dr. C. T. Brodnax (AFAL/TEA), Project Engineer.

This final technical report covers work conducted during the period February 1974 to December 1976 and was submitted by the authors on 1 March 1977.

Contributions to this report were made by R. F. Kolc, H. F. Inacker and P. N. Bronecke of the Missile and Surface Radar Division and W. F. Gehweiler and J. I. Pridgen of the Advanced Technology Laboratory in addition to the listed authors. Excerpts have been taken from the Phase I report, AFAL-TR-74-120 for continuity and completeness.

919 976L

Publication of this report does not constitute Air Force approval of the reports' findings or conclusions. It is published only for the exchange and stimulation of ideas.

Preceding Page Blank - Filmed

TABLE OF CONTENTS

SECTION PAGE

TABLE OF CONTENTS (CONTINUED)

SECTION PAGE

TABLE OF CONTENTS (CONTINUED)

SECTION PAGE

TABLE OF CONTENTS (CONTINUED)

TABLE OF CONTENTS (CONTINUED)

LIST OF ILLUSTRATIONS

FIGURE PAGE

LIST OF ILLUSTRATIONS (CONTINUED)

LIST OF ILLUSTRATIONS (CONTINUED)

LIST OF ILLUSTRATIONS (CONTINUED)

LIST OF TABLES

SECTION I

INTRODUCTION

This final report on Contract Number F33615-74-C-1077 describes results of Phases II and III covering the hardware fabrication and evaluation phase of the digital Programmable Fast Fourier Transform Linear FM Waveform Processor (PWP)*.

The PWP program features three principal developments; the step transform processing algorithm, floating point FFT processing and CMOS/SOS LSI technology. The step transform algorithm was studied and its realizability demonstrated under a pulse compression techniques study program(1) (F33615-72-C-1634). It features a sub-aperture processing technique for the digital pulse compression and expansion of linear FM waveforms. The algorithm was extended to sub-array processing of synthetic aperture radar (SAR) in Phase I of the PWP development program(2) (F33615-73-C-1275). Both SAR azimuth focusing and linear FM pulse compression can be processed with the same hardware configuration.

The objective of the hardware development effort was to construct a programmable pulse compression/expansion system for use in advanced linear FM radar and synthetic aperture processing applications. Processor goals included time bandwidth products programmable in the range of 100 to 1000, sidelobe levels less than -35 dB, and clock rates up to 10-12 MHz.

A functional diagram of the PWP system is shown in Figure 1. The figure indicates the major functional subsystems and the CMOS/SOS circuits which are used in them. Pipeline architecture is employed which permits input data to be processed in the system in real time.

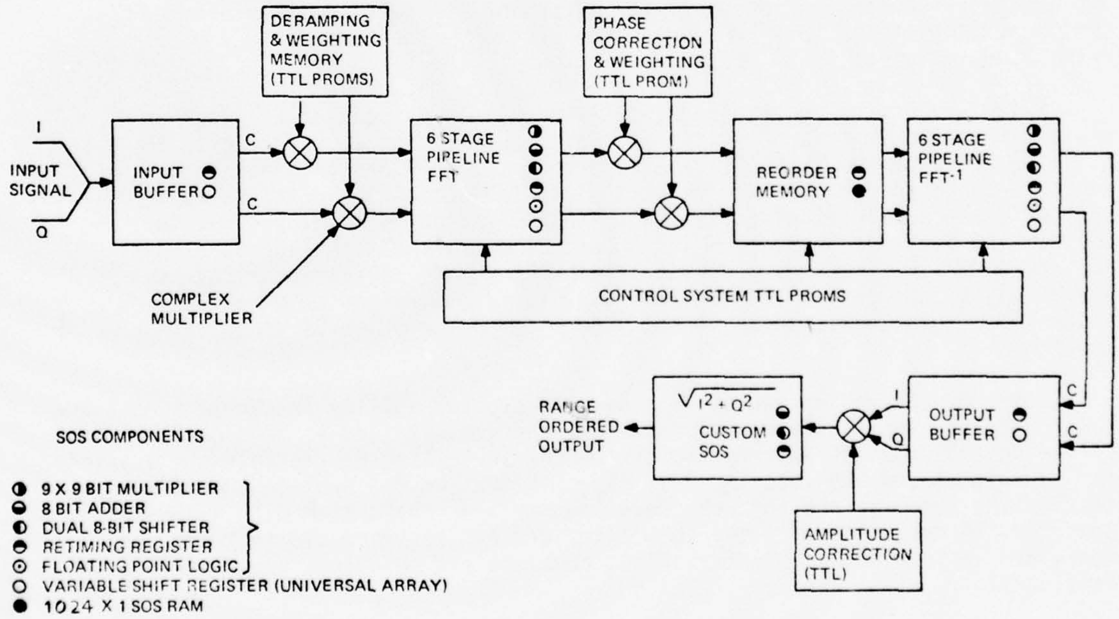


FIGURE 1. FUNCTIONAL DIAGRAM OF PWP SYSTEM

*The program was initially called the Programmable Waveform Generator (PWG) and was renamed during the final phase. The initials have been abbreviated to PWP in this report for convenience and to retain consistency with the previous designation.

Floating point architecture is used in the pipeline FFT's which are an inherent part of the implementation of the step transform. This maximizes the system performance level versus the number of quantization bits in the processor and hence also the hardware complexity. A quantization level of 8 bits plus sign for the in-phase (I) and quadrature (Q) components of the complex data words together with 4 bits in the characteristic was selected for the PWP. This quantization gives a performance level of about 50 dB in target dynamic range and places the mean-square error at the output relative to a peak signal level at less than -70 dB.

The CMOS/SOS design of the PWP makes maximum use of redundancy of functions, circuits and pluggable modules. All elements of the functional pipeline of the PWP are implemented in CMOS/SOS while the control system employs TTL because of the non-repetitive nature of the functions. The SOS technology permits the application of LSI and 10 MHz operating speed, with a system power consumption of about 460 watts. Five new custom SOS LSI designs are used for the PWP arithmetic operations. A sixth new SOS circuit uses an SOS Gate Universal Array approach to obtain a programmable shift register memory. The bulk memory requirements are met with an SOS random access memory. Thus, seven SOS LSI circuit designs constitute the bulk of the hardware requirements of the PWP. These are partitioned in six basic functional hybrid module types which house about 70 percent of the circuits employed in the PWP. Universal modules holding individually mounted CMOS/SOS or TTL circuits are used for clock drivers and the miscellaneous control functions.

Delays in the program primarily involving the CMOS/SOS circuit development and quantity fabrication together with attendant increased costs prevented implementation of the full PWP during the program. The forward and inverse FFT's were implemented and tested.

A summary of the program and results are given in Section 2. Section 3 reviews the step transform algorithm and its application to both linear FM pulse compression and synthetic aperture radar. Basic design, fabrication, and test concepts for the PWP are also given in Section 3. Detailed design and performance characteristics of the CMOS/SOS circuits developed for the program are provided in Section 4. A new LSI packaging technique developed for the CMOS/SOS circuitry and associated wiring and fabrication rules is described in Section 5. The CMOS/SOS circuits are implemented on eight module types whose functional, physical and electrical description is contained in Section 6.

Some innovative approaches to achieving programmability in signal processing control systems were developed for the PWP and are detailed in Section 7. Following the physical description of the PWP in Section 8, the software developed for the system and key subsystems is given in Section 9. Section 10 contains a description hardware and software developed to test the system with a PDP-11/20 computer. A description of the module tester, its hardware and software, is contained in Section 11. Test results, problems and solutions are described in Section 12.

SECTION II

PROGRAM SUMMARY

2.1 RESULTS - ACCOMPLISHMENTS

2.1.1 Signal Processing System Developments

The PWP program resulted in a number of key system related accomplishments in addition to the basic hardware developments. These are listed in capsule form in Table 1. First was the extension of the step transform algorithm to synthetic aperture radar during Phase I (2). The step transform algorithm has the distinct advantage of offering a moderate complexity solution to motion compensation in addition to providing a more efficient SAR hardware implementation.

TABLE 1
PWP SYSTEM DEVELOPMENTS

ITEM	FEATURES
Extension of Step Transform Algorithm to Synthetic Aperture Radar	Moderate Complexity Method for Motion Compensation
Waveform Generator Algorithm	Uses Same Hardware for Both Linear FM Pulse Generation and Compression
Development of Reorder Memory Concept	Minimizes Memory Size - Provides Programmable Addressing Technique to Minimize Address Sequence Storage
Partitioning for Modular Construction	PWP Processor Implemented with Only Eight Module Types, 6 Functional Module Plus Two Universal Types
PROM Based Control System for Programmable System	Minimizes Number of Circuits in Achieving Programmability for PWP Processor

Waveform generation capability is an inherent feature of a true dispersive delay function and the possibility of using the PWP hardware for this purpose was obvious. However, the actual implementation within the constraints of the hardware design was not straightforward. An extensive iterative computer procedure (Section 9.2) was necessary for the computation of the proper phase coefficients to produce the desired coherent output waveform.

A key feature of the step transform algorithm is that the total bulk memory storage requirement is, in theory, equal to the number of samples in the waveform being processed. In addition to the storage requirement, the bulk memory must reorder the data along successive time-frequency diagonals. Complicating the process is the inherent bit reversed data sequence which is fed to the memory by the FFT processor. Two methods were conceived and patented during the PWP program which can achieve the theoretical minimum in memory storage for a step transform processor. One of these methods uses a shift register technique which has the advantage of a very simple control mechanism, but the disadvantage of requiring large shift registers and their accompanying high clock power. The second technique, adopted for the PWP, uses a RAM based reorder memory. A simple, programmable memory address generation scheme was developed which minimizes address storage for the RAM memory.

In the step transform, as in many advanced digital signal processors, the complexity of the hardware is very high. Another typical characteristic of a real time digital signal processor is that many of the basic arithmetic functions are either repeated many times or are very similar throughout the system. Thus, the overall hardware complexity and cost can be minimized if this functional redundancy can be exploited by partitioning the system into a limited number of functional blocks. The design of the CMOS/SOS LSI circuits and modules in the PWP has achieved this end. Six functional module designs and two universal module designs used primarily for control are incorporated in the total module count of 198 in the PWP system.

The control system in the PWP requires changing inputs to the pipeline as the TW product varies. This was achieved in the PWP by extending the use of PROMs to a control system hierarchy. Successive stages of PROMs were used to give indirect addresses which maintained a simple reference or timing sequence at each pipeline FFT or processor stage. This approach minimized the total reference memory requirements and should be applicable to a general class of programmable processors.

2.1.2 CMOS/SOS LSI Circuit Developments

Earlier studies at RCA(7) had developed the special floating point FFT arithmetic concept and it was recognized that a high performance could be attained with only a quantization level of 9 bits floating point versus the 12 to 24 bits fixed point being employed by the industry for FFT processors. A quantization level of 8 bits plus sign could be handled within the developing state-of-the-art of CMOS/SOS LSI technology for the functional elements of a radix-2 pipeline FFT. The design, architecture and specifications for the various functional elements were incorporated in the designs of a number of LSI circuit development programs outside of the PWP program. The six LSI circuits specifically designed and fabricated for the PWP program are given in Table 2 with their contract funding source and main features. In addition to satisfying the particular needs of the PWP program, these circuits were designed, insofar as was practical, to have generally wide utility for other applications. All of the basic designs were made with the goal of at least a 10 MHz system clock rate. Functional circuits generally do not have reclocking registers incorporated on them and this permits construction of arithmetic functions of lower speed with a minimum number of circuits.

TABLE 2
CMOS/SOS LSI CIRCUITS DEVELOPED FOR PWP

	FUNDING	FEATURES
9 X 9 Multiplier	(TCS-001) - F33615-72-C-1291 (TCS-057) - PWP	8 Bits Plus Sign, Sign-Magnitude Multiplier With Optional 8 Bit Rounded Product
9 Bit Adder	(TCS-008) - N00014-73-C-0090 (TCS-065) - PWP	9 Bit One's or Two's Complement Adder With Overflow Detection and Compensation
Dual 8 Bit Scaler (TCS-016)	F33615-33-C-5043	Dual 8 Bit Position Scaler for Floating Point Applications and Other Binary Division
Retimer Register (TCS-015)	F33615-33-C-5043	18 Bit Reclocking Register With Complement Select
Floating Point Logic Control (TCS-017)	PWP	Floating Point Control for FFT Arithmetic Unit of Arbitrary Radix (Parallelism)
Programmable Shift Register (TCS-060-400B)	PWP	Highly Flexible Shift Register With Variable Length, Complementing Functions and Switched Delays. Total Registers= 38 Bits

The 9 x 9 multiplier (TCS-057) (8 bits plus sign) is the most complex LSI circuit of the group. It will provide either a full 16 bit output or an 8 bit rounded product. The multiplication is done in sign-magnitude so that inputs and outputs are of this form.

Functionally, the 9-bit adder (TCS-065) is unique in that it has built-in overflow detection and compensation. That is, if an overflow (carry) is detected the data can be automatically divided by two (shifted) to maintain the same number of quantization bits. This feature is particularly useful in floating point arithmetic where the overflow bit is then added to the exponent word. The adder can add numbers in either ones or twos complement form.

Conventional commercial scaler circuits or barrel shifters do not handle negative numbers. The ability to properly scale negative numbers is a characteristic of the dual 8 bit scaler (TCS-016). The scaler is also a convenient circuit for division by any power of two.

The retimer register (TCS-015) will reclock an 18 bit word. It is divided into two segments of nine bits and can be used to obtain the ones complement of the two nine bit input words.

The only circuit which appears to be specifically limited to pipeline FFT applications is the floating point logic array (TCS-017). It provides the desired floating point arithmetic control functions. It has a built-in capability to handle higher order FFT radices such as radix-4 or radix-8. These provide higher degrees of parallelism for higher speed processor requirements.

The programmable shift register was designed to incorporate all of the functions required in input buffer, output buffer, and FFT memory as the TW product of the PWP was varied. It has a total of 38 register stages with various switching, length change and complement controls to provide the desired functions.

2.1.3 PWP Hardware Developments

The PWP program achieved several significant hardware developments listed in Table 3. This was the first program to require a quantity of CMOS/SOS LSI circuits to be fabricated and implemented in a system. While several problems were encountered in this effort (discussed in Section 2.2), the overall effort can be judged as successful. Quantity parts were fabricated and implemented. The problems were either solved or solutions identified so that future efforts will be able to move forward in a much more confident and predictable manner.

TABLE 3
PWP HARDWARE DEVELOPMENTS

ITEM	FEATURE
Quantity Fabrication and Implementation of CMOS/SOS circuits	One of first systems employing large quantitv of CMOS/SOS circuits.
LSI packaging approach	Provides low capacitance interconnections, high density, ease of fabrication. Design rules were developed for packaging.
Functional Modules, 6 Types	Modules are designed for multi-function use and have wide application.
CMOS/SOS Clock Distribution	Individual clock drivers with equalized loads.
Module Test Facility	Computer controlled testing of modules at high clock speeds
System Test Facility	Designed for both system evaluation and hardware checkout.
Pipeline FFT, Programmable Length	Successfully completed construction and testing of pipeline FFT up to 5.2 MHz clock rate. Speed limited by a few low speed circuits.

CMOS/SOS LSI technology required the development of a new packaging technique. The basic requirements were small size, easy, reliable fabrication and low capacitance interconnections. These requirements have been met by placing the LSI circuits in leadless hermetic carriers which are mounted on a thick film ceramic substrate by reflow soldering. The leadless carrier approach solved the low fabrication yield problem projected with the use of a chip and wire-bond technique in addition to offering a low cost assembly operation. While the size is not as small as can be achieved with wire-bonds or beam lead attachment, the size is consistent with standard high density plug-socket combinations and until higher density plugs are available, the efficiency of the more dense packaging will be limited. The PWP program developed wiring and fabrication rules for the leadless carrier packaging technique.

The functional module complement which was developed for the PWP, fabricated and tested, is listed in Table 4. The first six of these modules provide discrete functions which are used to construct the processing elements required by the step transform algorithm. They are sufficiently complete functional blocks so that they may be applied to an extensive number of digital signal processing applications.

TABLE 4
PWP MODULES FABRICATED

MODULE	FUNCTION	NUMBER
Complex Multiplier	8 Bit Plus Sign Multiplication of Two Complex I and Q Data Words	9
Complex Adder	Addition and Subtraction of 8-Bit Plus Sign Words with Floating Point Characteristic. Includes Input and Output Scaling.	26
FFT Memory	Shift Register Memory for up to 6 Stage Radix-2 FFT. Adaptable to Input and Output Buffering Requirement of Step Transform Process.	22
Control Switch	Data Selector, 1 Bit x 8 Bit Complex Multiplier	17
Level Shifter	Shifts and Reclocks 18 TTL Inputs to SOS Levels	14
Reorder Memory	11 Bits x 1024 Word Memory	1
Universal SOS	1-48 Pin LSI Circuit 1-16 Pin Flat Pack	19
Universal TTL	4-16 Pin, 1-14 Pin DIP Circuits, Clock Drivers	50
	TOTAL	158

The control system used in the PWP is constructed with TTL logic because of the large number of MSI functions available for implementing the required controls. The clock distribution system also has a Schottky-TTL base and uses quad 75365 TTL to CMOS drivers as the final step of a clock distribution tree. This technique proved to be very successful.

The complexity of the functional modules in addition to the system itself prescribed the development of both a module and a system test bed. These systems are controlled by a PDP-11/20 computer and are capable of operating the unit under test at the full system clock rate. The system test bed has features allowing both system evaluation and step by step checkout of the pipeline hardware.

Finally, the circuits, modules, nest-backplane and test bed were successfully integrated and operated. The forward and inverse FFT were fabricated. Tests on the forward FFT provided zero errors with a variety of waveforms at speeds up to 5.2 MHz. The speed was limited by a few circuits whose speed was outside of the normal distribution. Two FFT stages of the pipeline operated at 8 MHz.

2.1.4 Software Developments

An extensive software library was developed during the PWP program; the major components, of which, are listed in Table 5. These software packages are described in the body of the report where applicable. The PWP system simulation is a refined version of the original simulation used to verify the performance of the step transform algorithm. The quantization levels and hardware logic operations are applied in an equivalent functional manner. The Hardware Logic Simulation is a complete hardware simulation of the PWP which uses the functional modules as building blocks. With this latter software, any system or portion of a system may be duplicated to obtain the p.ecise word patterns which the hardware will provide when operating correctly. This simulation was the basic reference tool in diagnosing the hardware operation during system debugging.

TABLE 5
PWP SOFTWARE DEVELOPMENTS

ITEM	FEATURE
PWP System Simulation	Simulation of all Hardware and System Functions
Hardware Logic Simulation	Bit by Bit Simulation of Logic Operation
Waveform Generator Coefficient Computation	Iterative Procedure for Determination of Phase Factors
Reorder Memory	Verification of Reorder Memory Hardware Design
Module Tests	7 Module Types
System Test	Permits System Tests and Hardware Checkout

The PWP system has a waveform generation mode and an iterative procedure was developed to determine the precise phase references in the hardware which would give a desired linear FM output waveform. The addressing sequence generation for the double-multiplex design of the reorder memory was completely simulated in order to verify proper operation before commitment of the hardware. Finally, extensive software packages were necessary for both the system test operation and the module tester.

2.1.5 Operating Results

The forward FFT was operated without error with a variety of waveforms at 1, 4 and 5 MHz. The length of the FFT was programmed to 16, 32 and 64 points and operated with zero errors in each case. The maximum speed for the forward FFT was 5.2 MHz and a portion was operated at 8 MHz. The speed was limited by a few low speed circuits as discussed in paragraph 2.2.2. The total power at 12 volts and 5 MHz averaged 1 watt per operating module.

Of 740 circuits implemented in the system, 597 were from the CMOS/SOS LSI group. Of these 33 had initial failures during module checkout, most of which were due to a packaging failure from ultrasonic cleaning. During operation 8, CMOS/SOS LSI circuit failures were identified.

2.1.6 PWP Program Patents

Patents applied for or awarded in the conduct of the PWP contract are listed in Table 6. In addition, the two key patents covering the step transform algorithm and floating point FFT process which preceded the PWP program, are also listed.

2.2 KEY PROBLEMS - SOLUTIONS

A number of problems arose during the PWP program which affected schedule and costs and which ultimately resulted in a reduction of the program scope to only cover fabrication and testing of the pipeline FFT's.

2.2.1 Circuit Design Problems

When a redesign of a custom LSI circuit is necessary, a very large schedule delay necessarily occurs since the design, artwork, mask generation, processing and test cycles must be repeated. Time was allotted during the program for a normal redesign cycle, but several problems occurred which can be classified as outside the normal.

2.2.1.1 Gate Universal Array - The CMOS/SOS Gate Universal Array (GUA) circuit for the PWP was the first SOS GUA to be fabricated. The initial circuit design used the bulk CMOS GUA masks and initial tests indicated that the expected SOS speeds were not being obtained. It was therefore necessary to redesign the basic GUA to provide the low capacitance interconnections required in an SOS design. Good speeds were obtained after the complete redesign.

2.2.1.2 Multiplier - The initial multiplier was designed under an RCAL contract with liberal, unverified circuit design rules. A very low yield was experienced at RCAL. This was the first CMOS/SOS circuit to be fabricated at SSTC (designated

TABLE 6
PWP PROGRAM PATENTS

<table>
<tr><th colspan="2">TITLE</th><th>INVENTOR</th><th>STATUS</th></tr>
<tr><td rowspan="2">Pre-PWP Program Patents</td><td>Digital Matched Filtering Using Step Transform Process</td><td>R. P. Perry</td><td>Patent Issued - 3987285</td></tr>
<tr><td>A Fast Fourier Transform Stage Using Floating Point Numbers</td><td>L. W. Martinson,
R. J. Smith</td><td>Patent Issued - 3800130</td></tr>
<tr><td colspan="2">Square Root of Sums of Squares Approximator</td><td>J. A. Lunsford</td><td>Patent Issued - 3858036</td></tr>
<tr><td colspan="2">Approximator for Square Root of Sum of Squares</td><td>J. A. Lunsford</td><td>Patent Issued - 3922540</td></tr>
<tr><td colspan="2">Data Processor Reorder Random Access Memory</td><td>L. W. Martinson</td><td>Patent Issued - 3943347</td></tr>
<tr><td colspan="2">Data Processor Reorder Shift Register Memory</td><td>R. P. Perry</td><td>Patent Issued - 3988601</td></tr>
<tr><td colspan="2">Dual Frequency Phase-Locked Loop Oscillator With Programmable Synchronization</td><td>J. A. Lunsford,
L. W. Martinson</td><td>Application Pending</td></tr>
</table>

TCS-001). The first run of the circuit was made with a mirrored mask set. This particular problem was due to the artwork being impressed on a mylar sheet and the mask operator could not easily tell in the absence of a script reference which side the print was on. Tests were made on the mirrored multiplier run which indicated that one or two operational chips were obtained. When a non-mirrored mask set was processed, zero yield occurred with repeated runs. It was finally necessary to redesign the multiplier with more recently established design rules to obtain good yield.

2.2.1.3 Adder Circuit - After the adder was completely tested and placed in the FFT arithmetic unit breadboard, it was discovered that a small logic oversight prevented it from operating properly in the radix-2 implementation. It was planned to correct the problem with a commercial CMOS/SOS circuit made by Inselek Corporation, but Inselek fell into bankruptcy before circuits could be obtained. Bulk CMOS circuits were not fast enough for the PWP application. The problem was finally solved by a redesign of the adder chip.

2.2.2 Circuit Processing Problems

2.2.2.1 Hydrogen Ion Contamination - During tests on the FFT memory module, unusual difficulties were encountered in that circuits would fail after operation for varying lengths of time. Normal operation was recovered after power was removed for several minutes. The problem was identified as a lot-dependent circuit instability problem at SSTC. Subsequent tests and analysis by SSTC identified a light hydrogen ion process contamination which occurred over a period of time. The primary cause of the contamination was determined to be a plasma etcher which had been placed in the pilot line several months before the problem was discovered. SSTC has now added a wafer probe test to detect an instability of this nature. All of the circuits made during that time period were replaced by SSTC at no additional cost. However, the replacement of the circuits caused a five month delay in reaching the point where quantity module fabrication could begin.

2.2.2.2 Low Speed Circuits - When the initial developmental CMOS/SOS circuits were tested, only a limited number (10-20) were available and these were generally processed in one or two process runs. The maximum and average speeds measured for these circuits coincided with predicted values. However, during the final system checkout, it was discovered that some of the system circuits were substantially slower than expected. This was found to be due to a process variable which caused a lower conductance level in the circuits. This parameter can be controlled. It should be noted that a circuit speed specification could not be accepted by SSTC when the PWP purchase orders were placed since CMOS/SOS circuit performance was not well established. Speed tests could not then be conveniently made due to instrumentation limitations. Speed tests will now be accepted by SSTC and yield is not expected to decrease by more than 10-20% with screening for speed.

2.2.2.3 CMOS/SOS 1024 Bit RAM - The reorder memory design selected used the first LSI component made commercially with CMOS/SOS by the RCA Solid State Division, a 1024 x 1 RAM. Initial start-up yield problems which have since been solved, delayed delivery of the high voltage units required for the PWP. These delays were a factor in the final decision not to implement the reorder memory.

2.2.3 Non-Circuit Related Problems

2.2.3.1 Ultrasonic Cleaning of Modules - During testing of newly fabricated modules, it was discovered that an ultrasonic cleaning operation caused wire bond breakage on certain circuit types. The cause was apparently due to the strong coupling to the metal-ceramic chip carrier and a resonance of long bonding wires. This cleaning method was eliminated and the problem disappeared.

2.2.3.2 Disc Breakage - The computer disc memory containing the PWP system software was found broken, and although the programs were printed-out, many of the details of the PWP system software had to be reconstructed. A back-up disc is now used in addition to programs under development being rolled out on magnetic tape at least at weekly intervals.

2.2.3.3 System Test Problems - The system test or debugging phase of the PWP program produced problems which were not generally unexpected. However, in addition to the normal, there were some problems which were somewhat unusual.

Over a period of time, a large number of control PROM bits became randomly programmed. Power supply transients and a possible short occurring when the test probe was plugged in were prime candidates for the cause, but a third possibility - a bad batch from the manufacturer has not been ruled out. All of the failed PROM's came from the same date code. The cessation of failures coincided with protection against power supply transients and depletion of the PROM's from the suspect date code.

Computational errors were found to be caused in some circumstances by excessive clock undershoot and additional damping was necessary on the clock drivers. Some time-dependent errors were found which were possibly due to a residual of the instability problem with the CMOS/SOS circuits. Backplane shorts appeared after a time which were due to the teflon insulation "creeping" at points where wires were drawn tightly around a pin.

2.3 RECOMMENDATIONS

The three phases of the PWP program extended over more than three years and covered system design, circuit design and hardware fabrication, assembly and test tasks. Many of the problems encountered could not have been foreseen at the inception of the program. However, it is worthwhile to note in retrospect what would be done differently given the lessons learned on the program or advances made on the program.

2.3.1 System - Waveform Generation

At a system level, the realizability of a step transform processor has been demonstrated and although this system includes digital waveform generation, this feature would not be generally recommended as a function. Developments in read-only-memories ROM's and programmable ROM's (PROM's) in recent years have made all but the very large waveforms easily stored on a small number of circuits. The total amount of storage required by the five PWP waveforms is 2822 words of 16 bits. These waveforms could be stored on six 1024 x 8 PROMs.

2.3.2 CMOS/SOS Circuits

2.3.2.1 Specification and Testing - All circuits should be specified with maximum propagation delays and rise times of key elements in addition to all of the functional and leakage tests. It also may be desirable to specify process related parameters (i.e., conductivity) where this is known to be significant. Circuits should be subjected to a 125°C dynamic burn-in. The burn-in should be dynamic to guard against the instability problem. A close contact should be kept with the circuit vendor by user quality control during the time of circuit fabrication and testing.

2.3.2.2 Memories - The use of shift register memories is a simple approach for storage and data manipulation, but as memories become large, they require high clock powers. To further complicate the situation, the PWP has dynamic shift registers which require two-phase clocks. Every bit in a shift register is active on every clock pulse. On the other hand, a random access memory (RAM) only has the decoding logic and a single word active on each clock pulse. The RAM, therefore, consumes much less total power per bit than conventional shift registers. (It should be noted that CCD memories extend the useful size of shift registers.) Because of their simple, flexible programmability and low power, a custom RAM design would be preferable for the PWP requirements in the FFT's and buffers over the current dynamic shift register approach.

2.3.3 Modules

Given a situation in which there were no size, technology or I/O pin limitations on the modules, a somewhat different module structure would result. There would be one add/subtract module per stage, one FFT memory module per stage, clock drivers on the modules, and CMOS/SOS control circuitry. Some advantage would be gained by more complete testing and characterization of modules in the same manner as individual circuits. A test procedure which would screen marginal thick film connections or potential shorts would be desirable. Some temperature cycling during test might accomplish this.

2.3.4 Nests - Backplane

The problem of shorting wires due to tightly turned corners on the backplane should be solved by using an insulation that does not "creep" under pressure, careful monitoring of the wiring process or point-to-point wiring.

2.3.5 Testing

In a technology development program such as the PWP, test procedures are generally not a concern in the early stages of development. However, the PWP and similar digital signal processing systems have such a high level of complexity that check-out and test features should be built-in to the basic design. The acronym generally used is BITE (Built-In Test Equipment). Although the BITE features can be expected to increase hardware costs up to 25 percent, much, if not all, of this cost may be regained during the life of the equipment.

A key part of an effective BITE function is the determination of the state of signals at key points in the pipeline. One way to achieve this would be to organize the retiming registers with an alternate serial shift mode. This would permit data to be frozen at a specified time and shifted out of the test point on a single line. A centralized microprocessor controlled facility could then aid in test signal analysis.

Preceding Page Blank - NOT Filmed

SECTION III

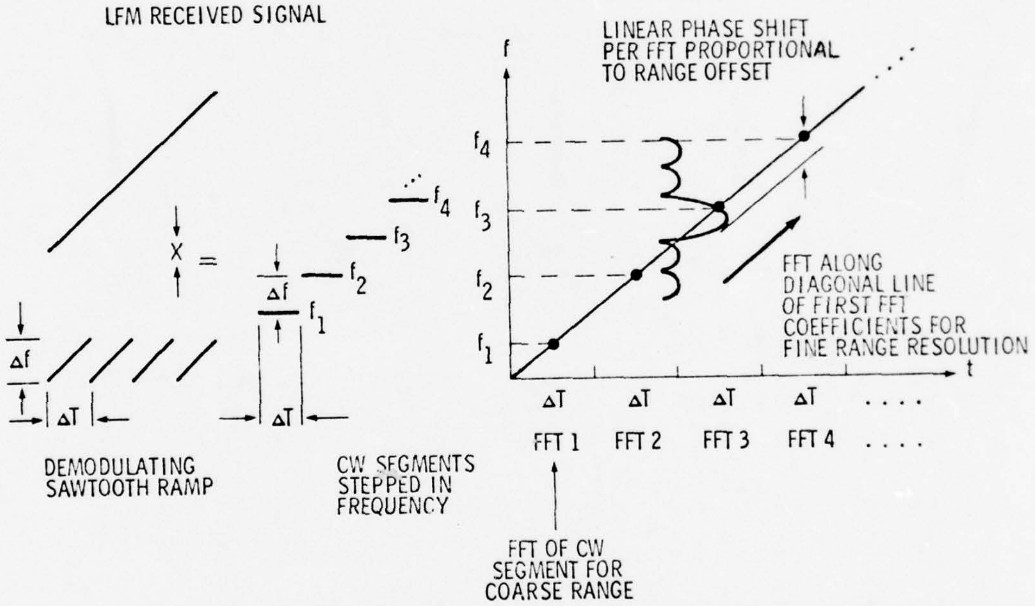
PWP SYSTEM DESCRIPTION AND PERFORMANCE

A complete functional description of the PWP has been given in Reference 2. However, for the purpose of continuity, the basic concept, key elements, and performance of the PWP are included here.

3.1 STEP TRANSFORM PROCESSOR CONCEPT

3.1.1 Linear FM Pulse Compression

The PWP has as its basis a new processing algorithm for linear FM waveforms called the step transform. The algorithm has been fully described and developed mathematically in previous reports (1,3) and only a summary of the technique will be presented here. A conceptual diagram of the step transform algorithm for linear FM pulse compression is given in Figure 2. The received signal is a linear FM waveform of length T and bandwidth W. This is sampled, A/D converted and multiplied by a linear FM sawtooth of length Δt and bandwidth Δf. The time-bandwidth product $\Delta t \cdot \Delta f$ of a single "tooth" is approximately equal to $\sqrt{TW}$. The resultant from this operation is a segmented CW waveform of about $\sqrt{TW}$ segments whose overall slope is equal to the slope of the original waveform.



LFM RECEIVED SIGNAL
X
=
Δf
ΔT
DEMODULATING SAWTOOTH RAMP
Δf
ΔT
f1
f2
f3
f4
CW SEGMENTS STEPPED IN FREQUENCY
f
LINEAR PHASE SHIFT PER FFT PROPORTIONAL TO RANGE OFFSET
f4
f3
f2
f1
FFT ALONG DIAGONAL LINE OF FIRST FFT COEFFICIENTS FOR FINE RANGE RESOLUTION
t
ΔT ΔT ΔT ΔT ....
FFT 1 FFT 2 FFT 3 FFT 4 ....
FFT OF CW SEGMENT FOR COARSE RANGE

FIGURE 2. STEP TRANSFORM ALGORITHM FOR LINEAR FM PULSE COMPRESSION

Each CW segment is in turn processed by an FFT to obtain its spectral coefficients. These coefficients can be considered as a coarse range estimate of the signal arrival. Each CW segment is processed by separate FFT analysis. If the start of the first demodulating ramp is in time coincidence with the received signal, the complex spectra will have zero phase along the diagonal traced by the changing frequency of the segments. There will be a linear phase shift along the diagonal proportional to the range offset. Thus a spectral analysis of the complex coefficients along the diagonal will produce a spectrum whose coefficients define the fine range resolution of the received signal.

The functional elements of the required processor for LFM pulse compression are shown in Figure 3. After the deramping function, each individual segment is passed into an FFT spectrum analyzer to obtain its exact spectral characteristics. The number of sample points in the deramping sawtooth is equal to the number of sample points in the input FFT aperture.

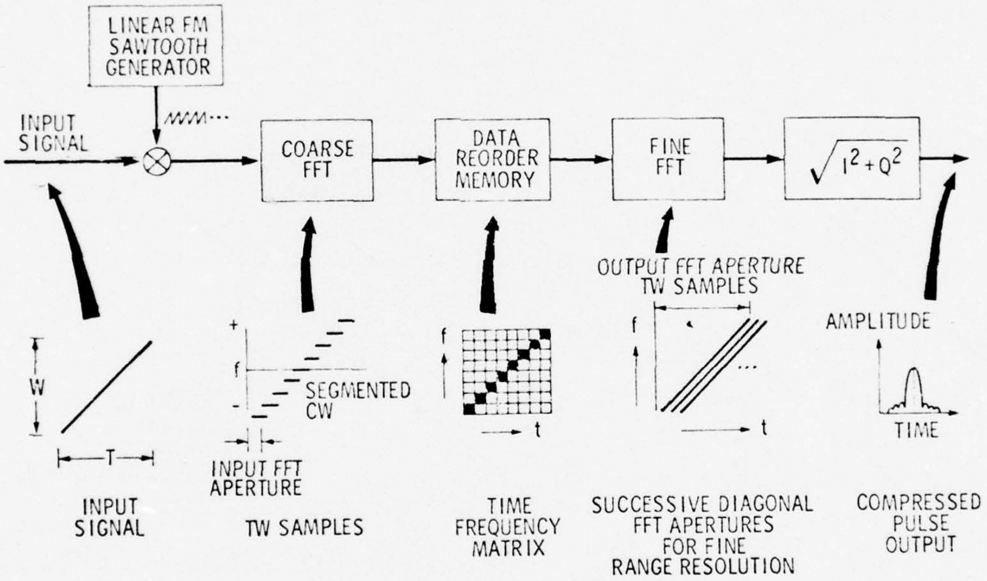


FIGURE 3. STEP TRANSFORM LFM PULSE COMPRESSION PROCESSING

Successive FFT analysis windows are stored in the data reordering memory which stores the data in a time-frequency matrix. Spectral data from a single received linear FM pulse will have amplitude peaks across the matrix beginning at a point corresponding to the coarse range of the target.

Fine range resolution is obtained by processing successive diagonals of the data in the time frequency matrix through the second FFT. The output of the second FFT gives the compressed pulse output with a resolution determined by the bandwidth of the waveform. Weighting is applied prior to entering the second FFT to reduce range sidelobes.

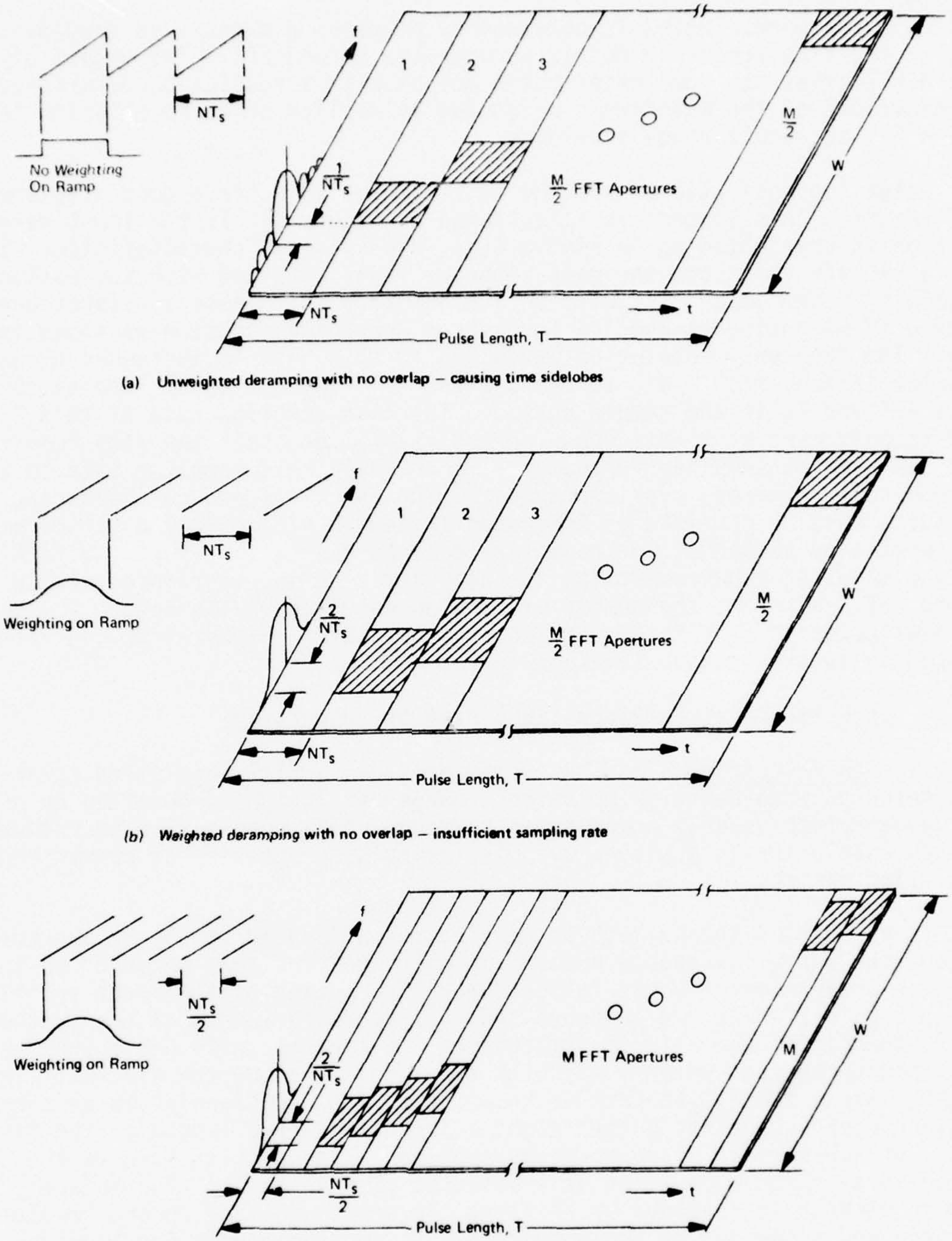
Weighting must also be applied at the input to achieve good sidelobe performance. This process is illustrated in Figure 4. If the input deramping function is unweighted as in Figure 4(a), the spectral characteristics will have a sin x/x shape and the peak sidelobe level obtained with successive diagonals in the second FFT will be a high -13.3 dB. These sidelobes are reduced if weighting is applied across the deramping function as shown in Figure 4(b). The frequency resolution bandwidth in this case is increased by about a factor of two, from $1/NT_S$ to $2/NT_S$ where N is the number of samples in the input FFT and T_S is the sample period. The time sampling rate of this $2/NT_S$ band is only made at a rate whose period is NT_S, one half the rate required to meet the Nyquist sampling frequency. To increase this sampling rate to the minimum rate required, overlapping deramping functions and corresponding FFT apertures must be provided as indicated in Figure 4(c). In the illustration, the repetition period of the ramps are decreased by a factor of two to $NT_S/2$ and the sampling requirement for the analysis band is, therefore, met in the second FFT. However, the number of samples required in the second FFT aperture are now increased by a factor of two from M/2 to M. Higher sampling rates may be used to further reduce sidelobe level.

3.1.2 Synthetic Aperture Radar (SAR) Azimuth Processing

3.1.2.1 SAR Principles - An SAR system provides a high resolution radar image of a selected area being illuminated. Range resolution is obtained by using a wide bandwidth usually accompanied by range pulse compression techniques. Azimuth resolution is achieved by pulse to pulse processing of common range resolution elements.

In performing the azimuth processing for a focused synthetic aperture system, the range (or phase) pattern of pulse returns are "matched" to that which an array element would follow due to the motion of the radar vehicle. The more pulses which are combined or the larger the length of the synthetic array, the higher the azimuth resolution. An aircraft will use a broadbeam side-looking antenna which moves with the aircraft along the aircraft line of flight. Since the flight path is known, the received signals can be coherently combined over successive pulses along a synthetic array length on the flight path. An appropriate phase shift is applied to these which focuses the synthetic array beam position at a selected azimuth angle. The adjacent azimuth element is obtained by shifting the array position up one resolution distance and a new set of phase functions is applied across the aperture. For a focused system, the pattern of the phases of an azimuth element is primarily quadratic (4,5). A quadratic phase function can also be represented as a linear FM time waveform.

A conventional method of digitally performing the required phase matching or convolution process is to employ a tapped delay line convolver. A problem with this implementation is that N phase weights are required for each digital



(a) Unweighted deramping with no overlap – causing time sidelobes

(b) Weighted deramping with no overlap – insufficient sampling rate

(c) Weighted deramping with overlap – low sidelobes, no aliasing

FIGURE 4. TIME WEIGHTING AND OVERLAP TO REDUCE TIME SIDELOBES AND TIME ALIASING

sample of the linear FM time waveform. As the number of samples, N, across the synthetic array increases, a large number of complex multipliers may be required to perform the phase weighting. This approach will result in a large amount of digital processing hardware for a digital SAR processor.

3.1.2.2 Subarray Processing - An approach which results in a considerable reduction of hardware while minimizing any performance losses is to process the synthetic aperture array as a set of subarrays each with $\sqrt{N}$ elements as shown in Figure 5. Each subarray is then focused toward the same azimuth element and the subarrays are then combined over the full aperture to obtain the high angular resolution. In performing this operation, it is also necessary to overlap subarrays to avoid the grating lobes that would result if the subarrays were placed end to end.

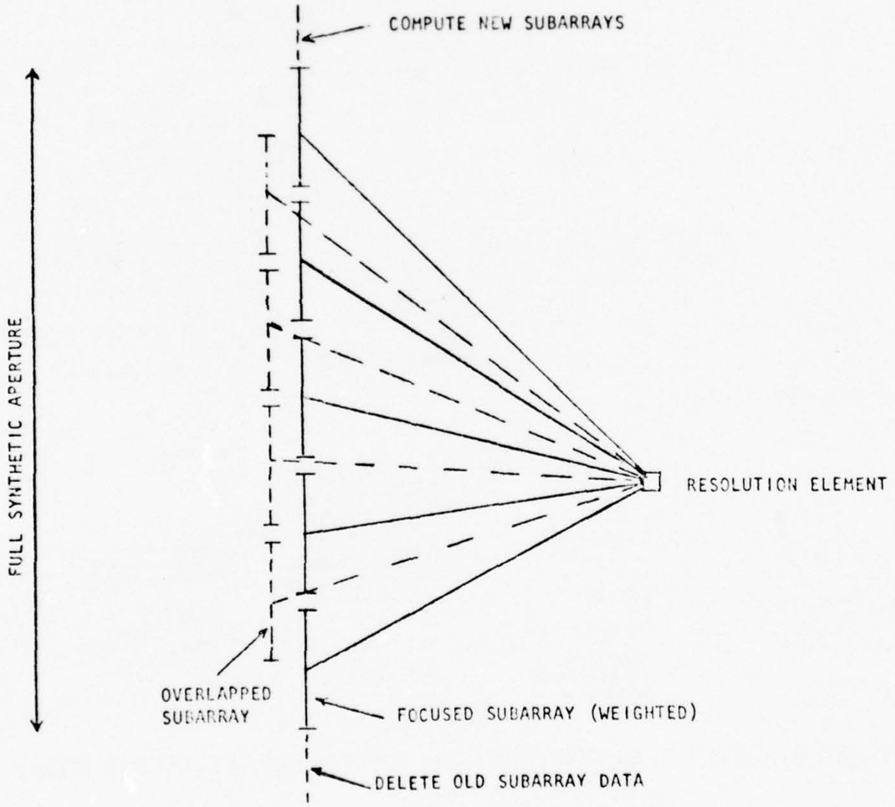


FIGURE 5. SYNTHETIC APERTURE AZIMUTH PROCESSING USING SUBARRAYS

As the full aperture is moved in space, each subarray will change its relative position in the aperture and will require a different beam position toward the focus point.

Since all subarray beam positions ($\sqrt{N}$) are used as it moves through the aperture, these beam positions can be computed as soon as the subarray data has been accumulated.

3.1.2.3 Application of the Step Transform to SAR - The step transform processoı architecture developed for linear FM pulse compression can be extended to SAR processing. The subarrays in SAR are analogous to the subapertures for pulse compression and the basic processing architecture is virtually identical as seen in Figure 6. This figure illustrates the application of the step transforı to a SAR telescope mode process.

An important problem in SAR processing is phase deviations across the aperture due to platform motion. The subarray approach permits compensation for this motion by incorporating a phase correction when combining the subarray outputs as shown in Figure 6.

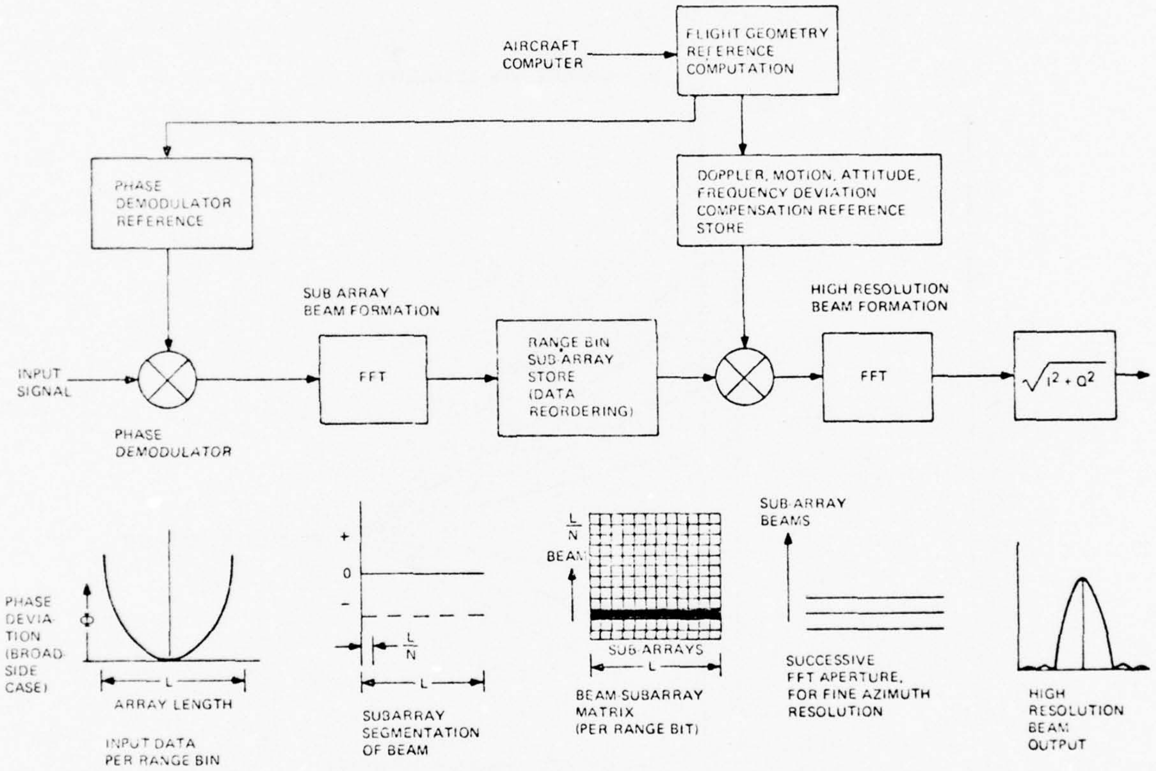

AIRCRAFT COMPUTER
FLIGHT GEOMETRY REFERENCE COMPUTATION
PHASE DEMODULATOR REFERENCE
DOPPLER, MOTION, ATTITUDE, FREQUENCY DEVIATION COMPENSATION REFERENCE STORE
SUB ARRAY BEAM FORMATION
HIGH RESOLUTION BEAM FORMATION
INPUT SIGNAL
FFT
RANGE BIN SUB ARRAY STORE (DATA REORDERING)
FFT
$\sqrt{I^2 + Q^2}$
PHASE DEMODULATOR
PHASE DEVIATION (BROADSIDE CASE)
Φ
L
ARRAY LENGTH
INPUT DATA PER RANGE BIN
+
0
−
$\frac{L}{N}$
L
SUBARRAY SEGMENTATION OF BEAM
$\frac{L}{N}$
BEAM
SUB-ARRAYS
L
BEAM-SUBARRAY MATRIX (PER RANGE BIT)
SUB-ARRAY BEAMS
SUCCESSIVE FFT APERTURE, FOR FINE AZIMUTH RESOLUTION
HIGH RESOLUTION BEAM OUTPUT

FIGURE 6. STEP TRANSFORM AZIMUTH PROCESSING (TELESCOPE MODE)

The synthetic aperture processing procedure using subarrays illustrated in Figure 6 can be summarized as follows:

° Focus data over short subarray time ($\sqrt{N}$ beam positions).

- ° Store subarray data over large array length ($2\sqrt{N}$ subarrays).
- ° Combine subarray outputs to form large array ($\sqrt{N}$ high resolution elements).
- ° Compute new subarray data entering large aperture.
- ° Drop old subarray data leaving large aperture.

3.2 SYSTEM PERFORMANCE - SIMULATION RESULTS

3.2.1 General

The PWP system performance was validated by computer simulation during Phase I of the program (2). The computer simulation implemented with a PDP-11/20 computer provides an exact mathematical model of the PWP. Hardware simulations are implemented by imposing quantization levels on mathematical operations to simulate exactly the logical operations.

The desirability of performing parameter tradeoff studies with simulation is obvious. Various system and hardware configurations can be evaluated, and over 350 various cases were simulated in studying the PWP. Computer simulation has been used concurrently with the detailed design and fabrication effort to verify subsystem and control system operation.

The PWP processor has three basic configuration or modes of operation:

1. Waveform Generation
2. Range Pulse Compression
3. Synthetic Aperture Azimuth Compression

Figure 7 is a simulated compressed pulse output (in range) from the PWP. The top trace is a representation of the transmitted uncompressed chirp pulse, and the bottom trace shows the compressed pulse output in a dB scale (20 dB/cm) for a time bandwidth (WT) product of 1183.

3.2.2 FFT Hardware Design

The hardware design of the low power CMOS/SOS PWP system 29 versus 31 now represents the results of extensive computer simulation studies of digital matched filters. These simulations performed by RCA have extended over a period of several years. Initial efforts were aimed at determining performance of pipeline FFT convolution matched filters.(6) These early studies indicated that matched filter performance with a fixed point FFT design was limited to a narrow dynamic range (less than 35 dB for an 8 bit plus sign system). The basic limitation of a fixed point FFT is due to the gain inherent in the FFT algorithm. As the signal is processed, its magnitude grows resulting in a "bit growth".

An alternate design for pipeline FFT has been developed by RCA (7) using a floating point approach. This floating point approach uses a complex word consisting of two fixed word size mantissas and a smaller word size exponent representing powers of two. The complex word consists of two mantissas: a real inphase (I) component mantissa; and an imaginary quadrature (Q) component mantissa. Each complex word has only one exponent value. Simply explained, the operation of the floating point hardware is such that overflows are detected

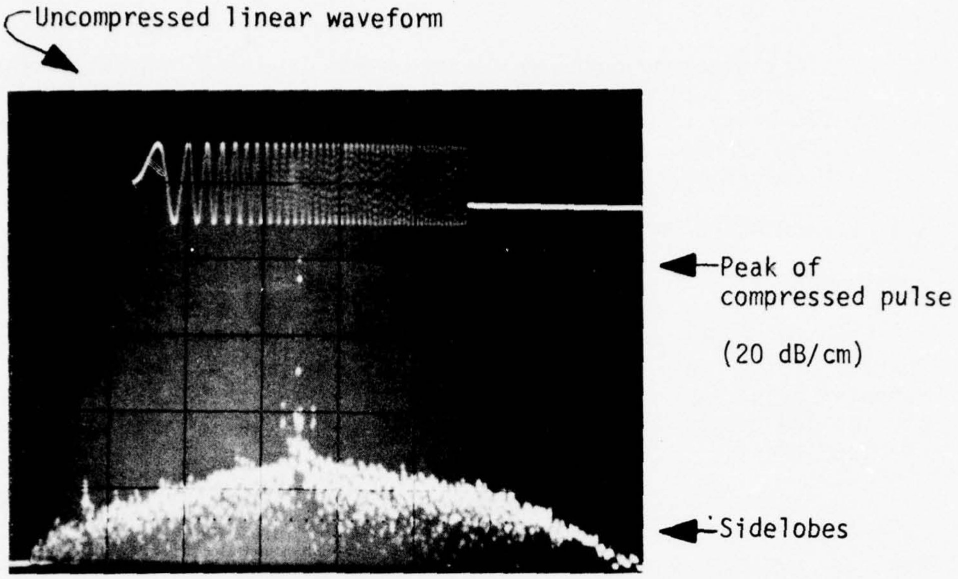


FIGURE 7. SYSTEM SIMULATION OF UNCOMPRESSED AND COMPRESSED PULSE
WT = 1183

after any arithmetic operation capable of generating an overflow. Detection of an overflow results in a right-shift of the complex word and incrementing the exponent. The floating point hardware also normalizes the data before an add or subtract by comparing the exponents of the two complex arguments and right-shifting the complex word with the smaller exponent while incrementing the smaller exponent until the exponents in both complex arguments are equal.

Figure 8 is a compilation of results from computer simulations of pipeline floating point FFT and shows the mean square error versus the number of bits in the mantissa. Although the simulations were for a convolution matched filter, the results are equally applicable to floating point FFT performance in a step transform configuration. The mean square error due to quantization was measured by generating a compressed pulse output using a simulator with a 16 bit floating point word and finding the errors by comparing that output with the outputs from simulations with varying quantization levels. Figure 8 shows that for an 8 bits plus sign quantization of the complex I and Q mantissa words, the mean square error of the sidelobes relative to the peak from a simple point target is less than -70 dB.

3.2.3 PWP Hardware Design

The objectives of a previous AFAL contract (Pulse Compression Techniques, Contract F33615-72-C-1634) included verification of the step transform algorithm

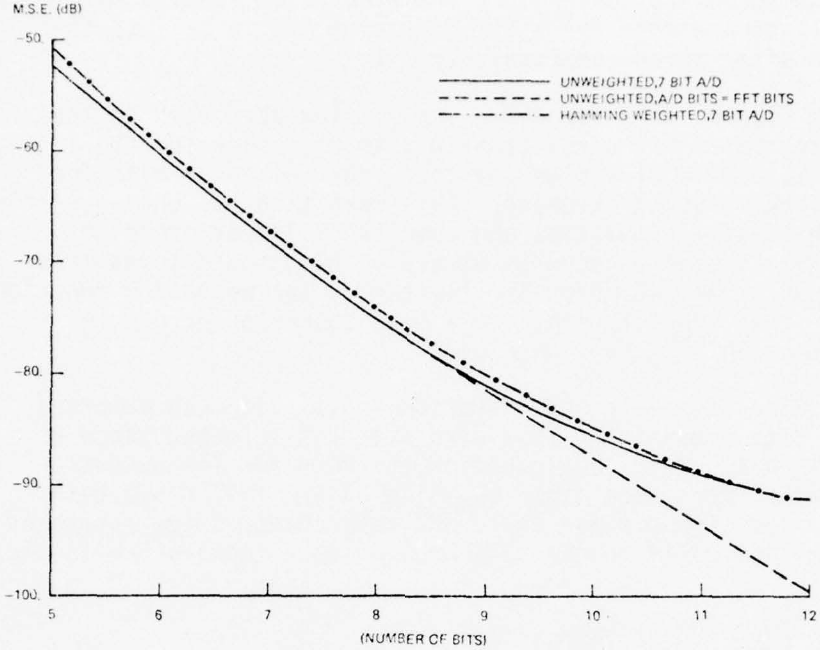


FIGURE 8. MEAN SQUARE ERROR VERSUS MANTISSA BITS IN FFT FOR SINGLE TARGET

and determination of hardware constraints through simulation.(1) Simulations of the step transform algorithm with a floating point FFT performed during the contract showed that peak sidelobes of about -6xn dB could be expected where n is the number of bits excluding sign in the mantissa of the floating point FFT.

All of the key features of the PWP hardware have been specified or verified by computer simulation. One of these features is the reorder memory. The specific function of the reorder memory is to transform the data samples from a column-row matrix sequence to a diagonal across the matrix. A technique for implementing the re-order memory using Random Access Memories (RAM's) has been developed and is discussed in detail in Section 9.3. In principle, this method uses the minimum memory size although addressing constraints and commercially available standard memory sizes will prevent achieving 100% efficiency.

Another feature of the PWP is the requirement for several reference memories (ROM's). These references are used to store FFT sine-cosine references, deramping functions, weighting, phase corrections, and data for waveform generation. The values for these reference memories have been determined from simulation and the precise binary values for the hardware are incorporated into the simulator. These stored reference values were printed out for programming the PROM's used in the hardware.

Based upon simulation results and required performance, the following word sizes were selected for the PWP hardware:

1. 7 Bit Plus Sign A/D - This represents current state of the art in high speed A/D conversion, and simulation results have shown quantization errors for 7 bit plus sign A/D to be less than -80 dB after pulse compression.

2. 22 Bit Complex Word for FFT - The complex word used in the floating point FFT consists of 8 bits plus sign for the I channel, 8 bits plus sign for the Q channel and 4 bits for the floating point exponent. This word size has been determined from extensive simulation and results in linear processor performance over a range in excess of 40 dB, sidelobes less than -35 or -40 dB which are limited by the weighting function rather than quantization, and a mean square error due to quantization of less than -70 dB.

3. 7 Bit Plus Sign Reference Memories - All reference memories are 7 bits plus sign. This word size was selected since 8 bits is a standard configuration for PROM's. The exception is the FFT reference memories which have a full 8 bit data word. The sign bit for these FFT references can be generated deterministically in the hardware for an effective 8 bits plus sign.

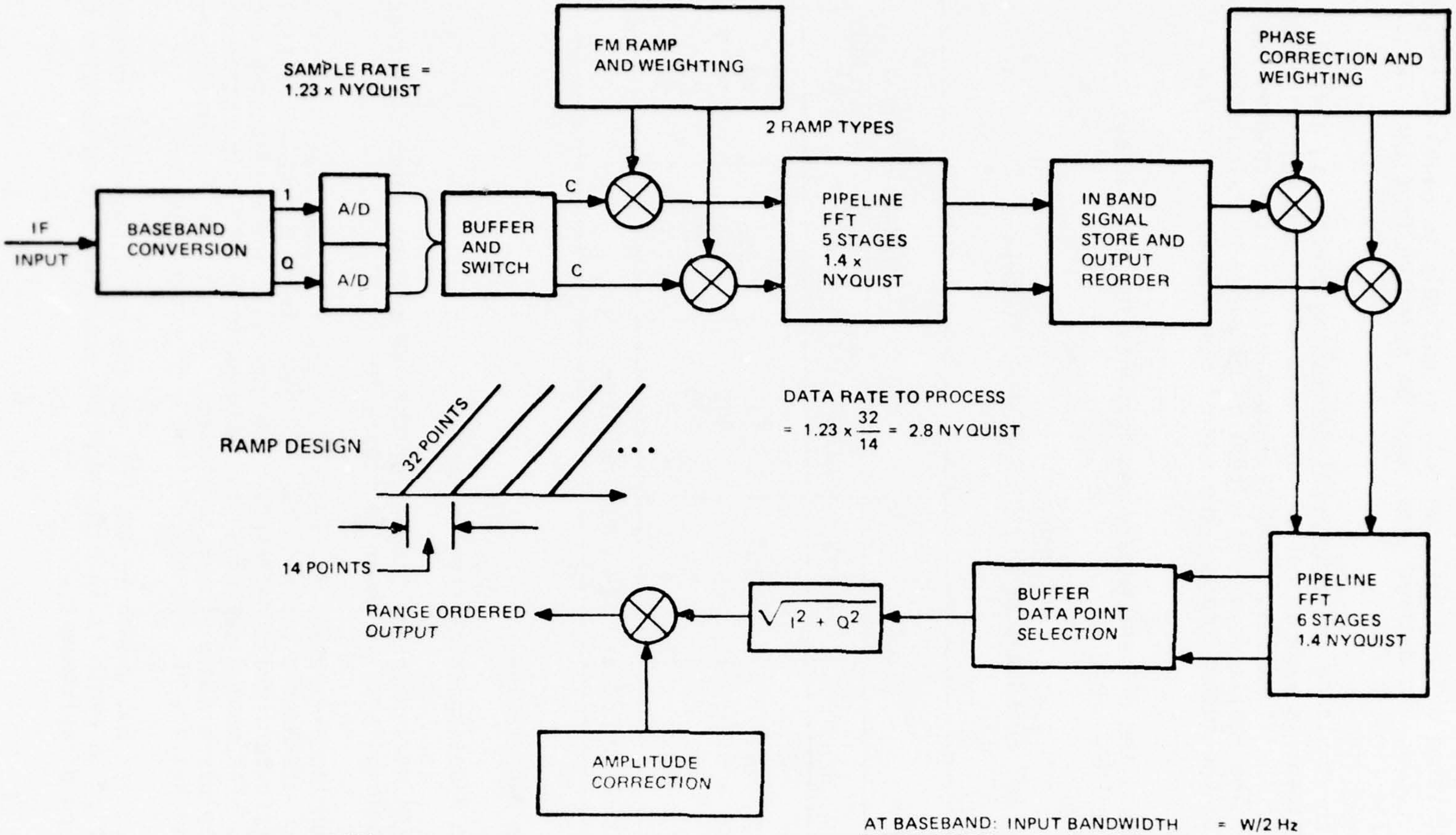
3.3 FUNCTIONAL ELEMENTS OF PWP SYSTEM

The primary functional elements of the step transform processor are highlighted in Figure 9 for TW = 592, which also indicates the data flow through the processor. The input is demodulated and sampled at baseband to provide in-phase (I) and quadrature (Q) signals. These baseband signals have bandwidth of W/2 Hz and the minimum sampling rate (Nyquist frequency) for them is W Hz.

In the PWP implementation, the I and Q signals are sampled at 1.23 times the Nyquist frequency. This factor is affected by the minimum sidelobe level required in the pulse compression system and the ratio of 1.23 is sufficient for -35 dB sidelobe levels.

After sampling, the input data is organized in overlapping signal apertures in the input buffer unit. The input signal is demodulated with a linear FM sawtooth of length equal to the first FFT aperture. To avoid aliasing of the time samples, successive ramps are overlapped to increase the sampling rate. Time weighting is applied across the input sample interval together with the FM demodulation to reduce the sidelobe levels in the frequency domain. An overlap ratio of 9/16 is used in the system. Thus, the net data rate necessary to process the input signal is then 1.23 x (16/(16-9)) = 2.8 times the input bandwidth.

Since radix-2 architecture is used in the pipeline FFT, two parallel input data streams are processed at the clock rate of the processor. The clock rate required for real-time processing in this case is 1.4 times the Nyquist frequency and N/2 sample intervals are required for an N-point FFT. The frequency coefficients from the first pipeline FFT are stored in a memory unit which also reorders the data for fine range analysis in the second FFT. Prior to insertion in the second FFT, a phase correction term is applied to the



NOTE:

AT IF: INPUT BANDWIDTH = W Hz
NYQUIST FREQUENCY = 2W Hz

AT BASEBAND: INPUT BANDWIDTH = W/2 Hz
NYQUIST FREQUENCY = W Hz (on I and Q)

FIGURE 9. IMPLEMENTATION OF STEP TRANSFORM FOR TW = 592

samples. The desired weighting function for range sidelobe reduction is applied in conjunction with the phase correction function.

The output range ordered data can only effectively be equal to the input sampling rate. Thus, the desired terms must be selected from the second FFT, a function of the final buffer.

Prior to thresholding and other post-processing functions, a final amplitude adjustment is made to correct for roll off of the weighting function applied ahead of the first FFT. This is incorporated in the complex multiplier at the output. The amplitude of the signal is obtained by calculating an approximation to the square root of the sum of the squares of the I and Q output samples.

Table 7 lists the FFT and memory requirements for the various modes designed for the PWP.

TABLE 7
PWP MODES AND MEMORY STORAGE REQUIREMENTS

TW PRODUCT	FFT STAGES		BUFFER MEMORIES	REORDER MEMORY (WORDS)	RAMP REFERENCE/ STORE
	FFT	FFT-1			
1183	6	6	280	2016	248
592	5	6	140	992	170
296	5	5	140	496	124
148	4	5	70	240	85
72,74,76*	4	4	70	120	62

*TW Products 72 and 76 are SAR Focusing Modes

3.4 PWP DESIGN ARCHITECTURE

3.4.1 Pipeline Floating Point Architecture

A more detailed functional diagram of the PWP system, which illustrates its modular and pipeline form, is shown in Figure 10.

The key functional element in the PWP processor is the pipeline FFT subsystem. Each arithmetic stage in the FFT consists of a complex multiplier, an adder, a subtractor, phase reference, a memory unit and a switching and control system. The memory switching is designed to permit calculation of the FFT with an absolute minimum of memory storage. A total of 2^N-2 words are stored within an N-stage FFT for calculating a 2^N point transform.

The problems of bit growth in the FFT and the non-linearities introduced by renormalization to avoid it in a fixed point processor are eliminated by using floating point arithmetic.

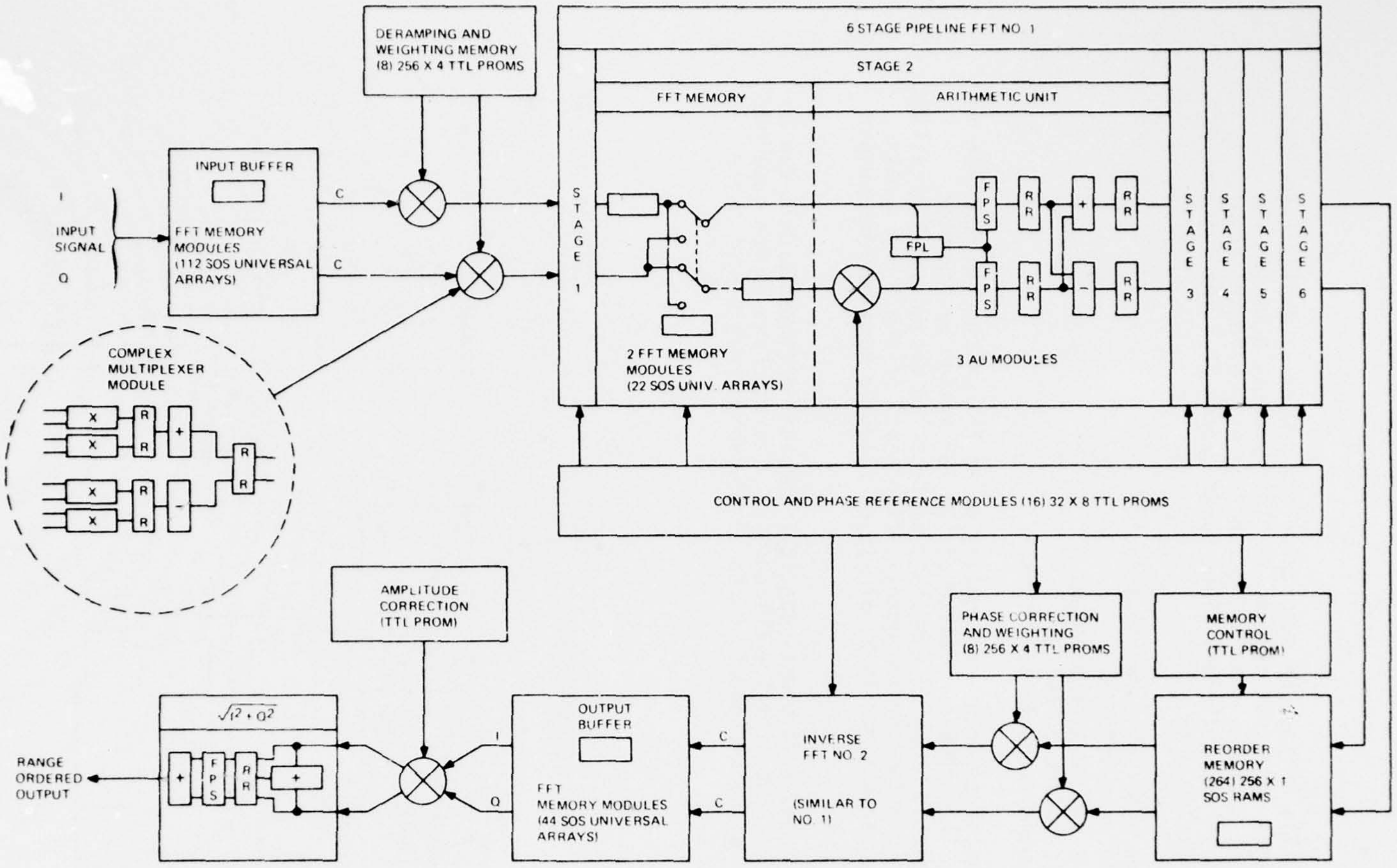

6 STAGE PIPELINE FFT NO. 1
STAGE 2
FFT MEMORY
ARITHMETIC UNIT
DERAMPING AND WEIGHTING MEMORY (8) 256 X 4 TTL PROMS
INPUT BUFFER
FFT MEMORY MODULES (112 SOS UNIVERSAL ARRAYS)
I
INPUT SIGNAL
Q
C
C
STAGE 1
2 FFT MEMORY MODULES (22 SOS UNIV. ARRAYS)
FPL
FPS
RR
+
−
3 AU MODULES
STAGE 3
STAGE 4
STAGE 5
STAGE 6
COMPLEX MULTIPLEXER MODULE
X
R
+
−
CONTROL AND PHASE REFERENCE MODULES (16) 32 X 8 TTL PROMS
AMPLITUDE CORRECTION (TTL PROM)
PHASE CORRECTION AND WEIGHTING (8) 256 X 4 TTL PROMS
MEMORY CONTROL (TTL PROM)
$\sqrt{I^2 + Q^2}$
RANGE ORDERED OUTPUT
OUTPUT BUFFER
FFT MEMORY MODULES (44 SOS UNIVERSAL ARRAYS)
I
Q
C
C
INVERSE FFT NO. 2
(SIMILAR TO NO. 1)
REORDER MEMORY (264) 256 X 1 SOS RAMS

FIGURE 10. DIAGRAM OF PWP SYSTEM WITH MODULAR ARCHITECTURE

The pipeline FFT's in the PWP step transform processor employ a special floating point process developed by RCA.(7) This process achieves the performance levels of true floating point function with hardware complexity much less than that of an equivalent fixed point processor. The features of the floating point system are:

- ° It eliminates need for normalization.
- ° Minimize size of multiplier.
- ° I and Q words have common exponent.
- ° Exponents are scaled up only.
- ° Maximum exponent required is 4 bits.

A block diagram of the radix-2 floating point FFT arithmetic function is given in Figure 11. A hardware modeling of the PWP system by computer simulation was completed prior to the start of detailed design and fabrication. This showed that an 8 bit plus sign quantization level of the I and Q data words gave a mean squared error level of -70 dB relative to a maximum output signal level.

If the exponent has sufficient capacity, the same arithmetic unit can be used at every stage in the pipeline FFT design. The exponent is not expected to go beyond 4 bits with TW products up to 16,000 which covers most pulse compression cases.

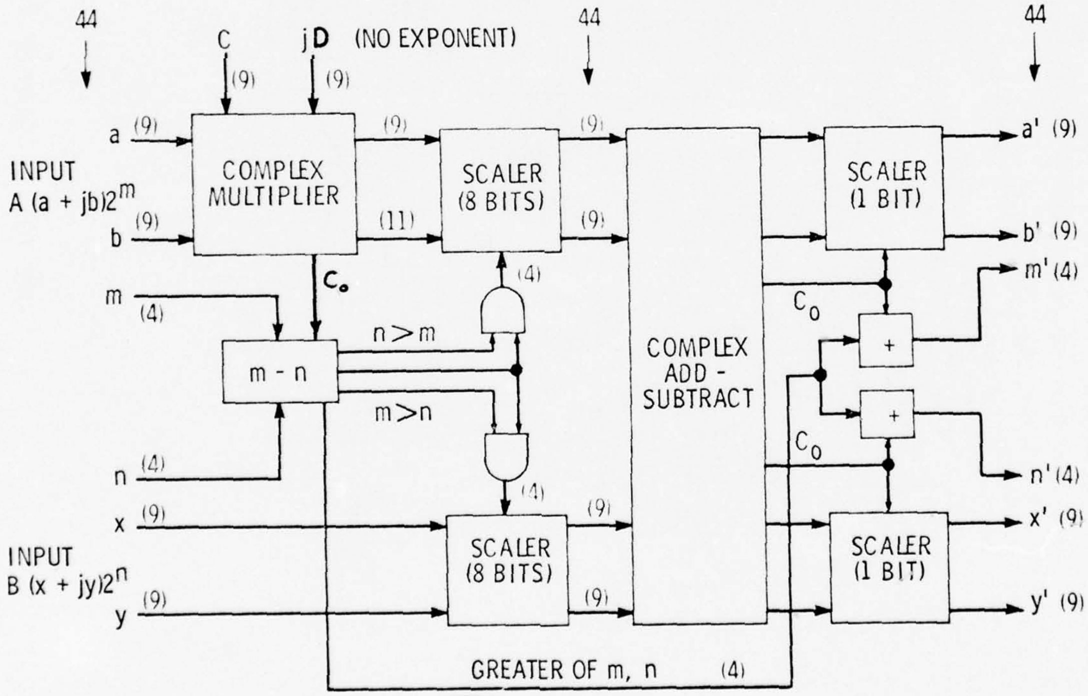


FIGURE 11. FLOATING POINT ARITHMETIC FUNCTIONS

3.4.2 CMOS/SOS LSI Circuit and Modular Partitioning

The principal objective in the design of the PWG has been for modularity of functions with a minimum of LSI CMOS/SOS designs.

In addition to the repetitive application of a limited number of CMOS/SOS components in the PWP, further repetition in the system is achieved by employing a limited number of functional module designs. The modular approach is illustrated in Figure 10. A single memory module serves the requirements of the input and output buffers and the FFT memory. The arithmetic unit of an FFT stage consists of 3 modules, a complex multiplier and 2 floating point adder/subtracters. In addition to these modules, a control/switching module, a random access memory module, a level shifter module and universal TTL and SOS modules are incorporated in the system. The universal modules are used principally in the TTL control area, for data delays and the $\sqrt{I^2 + Q^2}$ function which is only required once in the system.

The programmable requirements of the system impose a number of mode controls which vary specific hardware operations of a number of the subsystems. These include:

- ° The input buffer size varies as does the control timing sequence.
- ° The deramping reference changes in length and form.
- ° The forward FFT changes length and the FFT sine/cosine reference sequence changes.
- ° The reorder memory size varies together with its control sequence.
- ° The output weighting function varies.
- ° The length of the second FFT changes together with the sine/cosine reference sequences.
- ° The output length and control sequence changes.
- ° The phase reference functions and in some cases the length of the second FFT change when switching from the waveform generation to matched filtering mode.

CMOS/SOS large-scale integration (LSI) circuitry is used in the main pipeline portion of the PWP system with all control signals provided by TTL devices. Six of the seven LSI circuits have been designed specifically to the PWP performance requirements. However, they possess general utility since the floating point logic chip is the only circuit whose application is limited. The circuits are partitioned in the system on functional hybrid modules as indicated in Figure 10. For example, a complex multiplier containing four real multipliers and two adders is contained on a single 1.7-inch by 5.6-inch module.

The input buffer of Figure 10 is designed to provide the overlapping apertures of the input data. Using the Gate Universal Array (GUA)* as the basic building block memory modules are assembled to provide this function. Each individual circuit provides up to 32 bits of shift register delay together with the required switching functions of the FFT and buffers. The input buffer is sized by the maximum TW product to be handled. A total of seven 32 bit memory word delays must be provided with 16 bits per word. This requires 112 universal array chips mounted on 14 modules. The memory modules themselves have a capacity for up to 11 bits to provide the FFT memory function for a full 22 bits with two modules per FFT stage. The configuration for the FFT memory is indicated in Figure 10 in the expansion shown of the second stage of the first FFT.

*The Gate Universal Array is a special CMOS/SOS LSI circuit on which a single metalization pattern is developed for interconnection of the circuit devices.

Three modules are employed in a standard arithmetic stage of the FFT. In addition to the complex multiplier, two complex adder modules perform the floating point addition and subtraction, each one operating on the I or Q data channel. A floating point adder module consists of a floating point scaler, two adders, a floating point logic array and retiming registers.

From the first FFT, the data is fed through two complex multipliers, which perform the phase correction and range sidelobe weighting, to the reorder memory discussed in more detail in the next paragraph.

The output buffer uses the same memory modules as the FFT and input buffers with the control inputs set for the output buffer function. Four modules house the output buffer.

3.4.3 Reorder Memory

The step transform processor requires a reordering of the samples out of the first FFT before they are fed to the second FFT. Specifically, the data samples must be transformed from a column-row matrix sequence to successive diagonals across the matrix. The general form of the matrix for a unit slope diagonalization is shown in Figure 12. The samples in each column correspond to the output coefficients of the first FFT. Successive columns are given for each FFT aperture processed. The aperture for the second FFT along the first diagonal is completed when sample number N^2 is received.

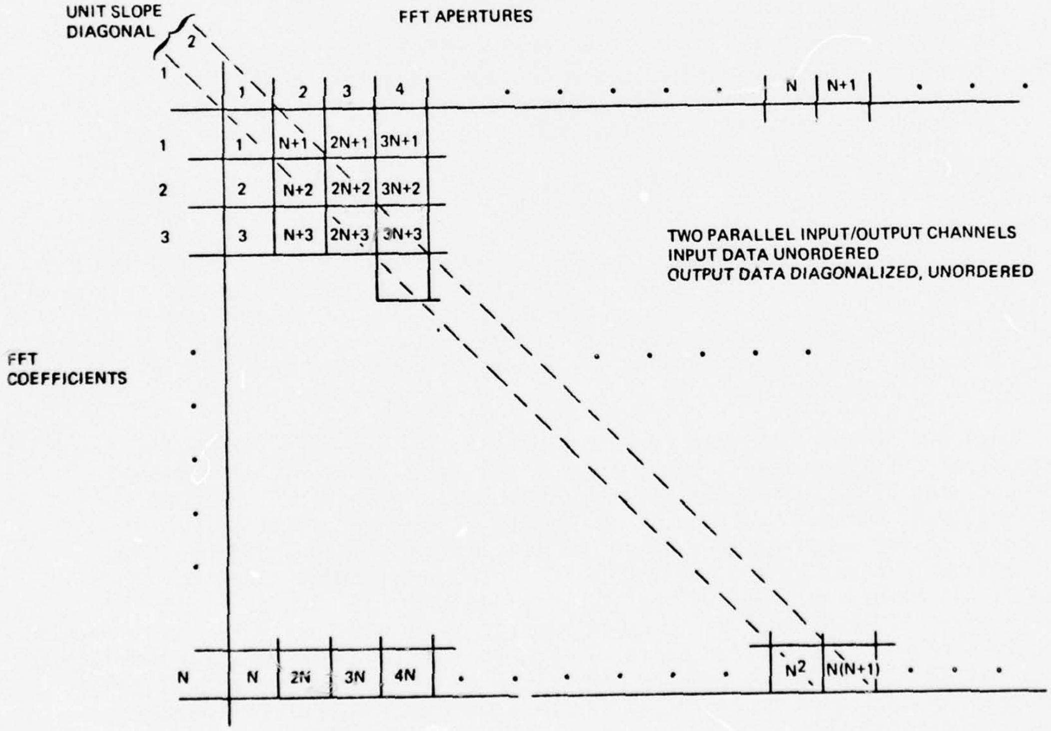


FIGURE 12. UNIT SLOPE DIAGONALIZATION

However, if the system is timed properly, data for the first diagonal can be fed to the second FFT commencing at the start of the N^{th} input FFT aperture. The N^2 sample is then computed just as it is required at the input to the second FFT. In addition, it is not necessary to store the data words to the left of the next diagonal to be processed. Therefore, the minimum storage requirements of the diagonalizing memory (for unit slope) is set by the triangular matrix bound by N FFT coefficients and N-1 FFT apertures. The minimum storage is N(N-1)/2 words.

The following factors complicate the memory design:

- ° The data received from the first FFT is in an unnatural (bit-reversed) sequence.
- ° The data fed to the second FFT must also be in a bit-reversed sequence.
- ° The input and output are two parallel data streams corresponding to the Radix-2 process.
- ° The reorder memory must handle additional slopes (TW products).

Although the reorder memory can be implemented using a shift register technique (Reference 2), the widespread developments in large capacity random access memories (RAM's) for computer application made their use more attractive for the PWP.

Operation at 10 MHz with large capacity current state-of-the-art RAM's can be achieved if two memory units are alternated, one reading while the other is writing. The basic form of the reorder memory system for the PWP is, therefore, as shown in Figure 13.

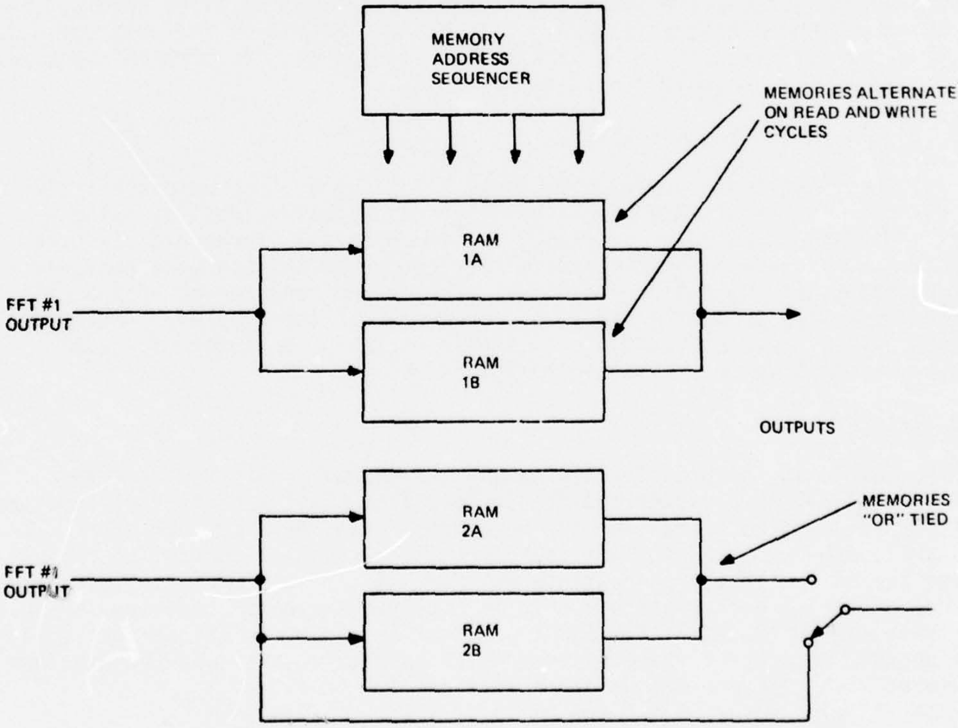


FIGURE 13. DIAGONALIZATION WITH RANDOM ACCESS MEMORIES

In the diagram of Figure 13, the data from the first FFT is alternately read into one of two RAM's. On alternate cycles, data is read out from a given RAM to the second FFT. In the actual PWP design, this concept is extended to a double multiplex form comprising a total of eight memory units described in Section 7.5.

3.5 FABRICATION, ASSEMBLY AND TEST CONCEPTS

3.5.1 LSI Packaging

The CMOS/SOS LSI circuitry imposed system packaging requirements which could not be met by conventional packaging techniques. To achieve maximum system speed, the circuit interconnections require low capacitance. Fifteen pf total capacitance was specified for the PWP. In addition, the large number of pins, up to 48 per package, resulted in typical modules having 300 to 400 interconnections.

An inexpensive, reliable method was required for assembling the IC's. Conventional chip-wire bond techniques had low probability of success because of the immaturity of the SOS technology and the large number of interconnections. During the initial phase of the program, efforts were directed toward achievement of beam-lead interconnections for the LSI circuits. However, it became apparent that beam-leads technology was not mature enough as applied to CMOS/SOS to be useful on the PWP program.

The technique which held the most promise and which was ultimately used on the program was to place the LSI circuits in leadless carriers manufactured by 3M Corporation under the tradename "Alsipak". The use of the chip carriers permitted the LSI circuits to be packaged and tested in a form which maximized the probability of assembling working circuits.

3.5.2 Module Design and Assembly

The functional modules which employed a 1.7" by 5.6" ceramic substrate were designed in the RCA MSRD Applicon ® automated design facility using a set of CMOS/SOS wiring rules developed for the program. These are discussed in Section 5. Standard ceramic hybrid fabrication techniques were employed for the substrate, but the art of attachment, repair and replacement of the chip carriers by reflow soldering had to be developed for the program. The results of these efforts have produced a new, viable packaging technique for LSI circuits which is dense, reliable and low in cost.

3.5.3 Test Programs

The complexity of both the system and individual modules imposed the requirement for both a system and module test facility. Both of these systems built for the program operate under the control of a PDP-11/20 computer. The basic philosophy employed in them is to provide a computer controlled capability to test the unit with any appropriate waveform at its full operating clock speed. To achieve this, a buffer storage unit is employed which can be loaded or unloaded asynchronously by the PDP-11/20 and input test waveforms to the unit at full clock speed. Details of these systems which have extensive software developments associated with them are discussed in Sections 10 and 11.

3.6 PROGRAMMABLE FFT LFM WAVEFORM PROCESSOR (PWP) SPECIFICATIONS

Function: Linear FM pulse compression and expansion, synthetic aperture radar processing.

Waveform Generation and Pulse Compression Parameters

WT	Time Sidelobe Weighting
1183	40 dB Taylor
591.5	40 dB Taylor
295.75	40 dB Taylor
147.87	35 dB Taylor
73.9	35 dB Taylor

Synthetic Aperture Processing Parameters

WT	Sidelobe Weighting
73.9	35 dB Taylor
71.9	35 dB Taylor
75.9	35 dB Taylor

Signal Bandwidth: Goal of 8.1 to 9.7 MHz

Processor Clock Rate: Goal of 10 to 12 MHz

Target Dynamic Range: >40 dB

Input Quantization: 7 Bits Plus Sign, I and Q

Processor Quantization: 8 Bits Plus Sign I and Q and 4 Bits Mantissa

FFT Processors: Pipeline, Radix-2 Floating Point Arithmetic. Two 64 Point FFT's Variable to 32 and 16 Points.

Hardware Technology: CMOS/SOS LSI, (TTL Control)

Packaging: Hybrid Modules, 80 Pin Connectors

Power Dissipation: About 300 Watts

Volume: <1.5 cu. ft. excluding power supplies and mounting cabinet.

Mean-Square Error Level: <-70 dB

Preceding Page Blank - NOT FILMed

SECTION IV

CMOS/SOS CIRCUIT DEVELOPMENTS

The six CMOS/SOS LSI circuits whose designs have been established by the PWP architecture are tabulated in Table 8. The design and initial fabrication of four of these devices was supported under other contracts as shown in the table. In addition to these six CMOS/SOS LSI circuits, a seventh is used, the 1024x1 CMOS/SOS RAM used in the reorder memory.

TABLE 8
CMOS/SOS CIRCUIT DEVELOPMENTS USED IN PWP

SOS CIRCUIT	FUNCTIONS IN PWP	DEVELOPMENT CONTRACT NO.
1. 8x8 Bit Plus Sign Multiplier	FFT vector rotation, deramping and weighting, phase correction and time sidelobe weighting (all above are complex multiplies)	F33615-72-C-1291
2. 9 Bit Adder	Complex multiplication and complex addition in FFT. Addition in $\sqrt{I^2+Q^2}$ approx.	N00014-73-C-0090
3. Dual 8 Bit Position Scaler	Floating point in FFT. Shift in $\sqrt{I^2+Q^2}$	F33615-73-C-5043
4. Retimer Register (18 Bits)	Reclocking of data in pipeline. Delay elements. Complementing function.	F33615-73-C-5043
5. Floating Point Logic	Floating point control in FFT	F33615-74-C-1077 (PWP)
6. Programmable Shift Register Memory	FFT memory, input buffer, output buffer	F33615-74-C-1077 (PWP)

This section discusses some early concerns with manufacturing techniques and standards which have evolved to an Ion Implantation Process with 0.25 mil channel length. A severe fabrication problem occurred midway in the program due to hydrogen ion contamination which created circuit instability. Most of the CMOS/SOS circuits had to be replaced due to this problem. This section also outlines the design and performance results of each of the circuits used in the PWP.

4.1 CMOS/SOS FABRICATION METHODS AFFECTING SPEED

4.1.1 Background

The initial development efforts on CMOS/SOS LSI circuits from which the baseline performance specifications for the PWP were drawn assumed the deep depletion (DD) fabrication process and 0.25 mil channel lengths in the LSI circuits. These initial assumptions were based on results then obtained by joint development results of the RCA Advanced Technology Laboratories (ATL) in Camden, NJ and RCA Research Laboratories (RCAL) in Princeton, NJ. Coinciding with the circuit development efforts had been research in manufacturing techniques and design rules by the RCA Solid State Technology Center (SSTC) in Somerville, NJ. The responsibility of manufacture of the PWP circuits rested at SSTC. Early in Phase II of the PWP program, SSTC design rules and manufacturing policies, based upon fabrication and yield results, established the double epitaxial (DE) process and 0.3 mil channel lengths as standards for quantity fabrication. These differences from the original PWP circuit design assumptions were projected to decrease the potential operating speed of the PWP by 10% for the DE process and up to 20% more for the 0.30 mil channel length.

An additional problem arose in that high voltage (15 volt) leakage breakdown was observed on test LSI chips fabricated at RCAL by the DD process. Operation at a lower voltage than the 15 volts assumed for the PWP was projected to cause a further speed reduction of 20% or more. These performance related process problems were of great concern for the PWP and a substantial effort was made to insure that they would be solved.

4.1.2 *Deep Depletion Versus Double-Epi*

The possible objective of employing the DD process for the PWP circuits evolved to a number of conditions as the program proceeded.

- The circuits could be made with the DD process, but no guarantees of performance or cost could be obtained.
- A pilot line was established for DE, guaranteeing costs.
- High voltage leakage was apparently not exhibited in initial tests of the DE process. This had a greater impact than DE versus DD.
- Ion implantation studies were conducted as part of the Manufacturing Methods Program at SSTC which provided another alternative to the standard DE and DD processes (8).
- Comparative measurements of a test chip fabricated with both the DD and DE process indicated no appreciable difference in speed. In addition, the DE measurements were faster than predicted; this was due in part to conservative circuit design assumptions.

Based upon the foregoing results, it was felt during the initial months of Phase II that the DE process would probably be satisfactory for the PWP. Furthermore, the possibility of obtaining circuits based on the ion implantation work had the potential of giving faster circuits.

Circuits tested for the PWP were processed at SSTC by the DD, DE and the I^2N/N ion implantation process which is an improved deep depletion process.

Results obtained on the adder circuit indicated that the 0.25 mil channel length is much more important than the DD process. Propagation delays of 0.25 mil channel DE circuits were from 14% to 17% faster than 0.30 mil DD circuits for operating voltages above 10 volts. Results from the ion implantation work made this DD process acceptable and most of the quantity circuits made for the PWP were processed in this manner.

4.1.3 Channel Length

In order to provide opportunity for SSTC to make comparative manufacturing tests for 0.25 mil and 0.30 mil channel lengths, all PWP circuit designs were made with both conditions. This meant providing complete mask sets of both types. With the requirement of a DD and DE option, a total of 4 mask sets were required for the custom circuits. This, of course, also provided the means by which the quantity circuits could be fabricated with any condition. Yield and reliability results with 0.25 mil channel have been satisfactory.

Tests were conducted on a 7 stage binary counter produced with 0.25 mil channel lengths. With conservative design rules as employed on the PWP circuits, very little yield reduction was observed between a 10 volt and 15 volt functional test. Recent reliability results using the more conservative design rules are given in Table 9 which indicate only one logic level failure after 383,500 device hours at 10 volts and 125°C.

TABLE 9
CMOS/SOS LIFE TEST SUMMARY (7-STAGE BINARY COUNTER)

LOT NO.	QTY.	HOURS @ 125°C	FAILURES	ACCUMULATED UNIT HOURS
312	10	2,000	0	20,000
319	10	2,000	0	20,000
324	15	2,000	0	30,000
331	19	2,000	0	38,000
335	20	2,000	0	40,000
348	16	2,000	0	32,000
372	9	2,000	0	18,000
383	9	2,000	0	18,000
389	17	2,000	0	34,000
407	17	2,000	0	34,000
427	17	2,000	0	34,000
453	5	2,000	0	10,000
498	8	2,000	1 @ 1000 Hours	15,000
943	27	1,500*	0	40,500
TOTALS	199		1	383,500

* Still on Test
MTBF (60% Conf. Level) ≃ 200,000 Hrs. @ 125°C, 10 Volts

The leakage tests on the adder comparing 0.25 mil DE with 0.30 mil DD indicate a higher leakage with the smaller channel length. However, this does not generally cause a significant increase in total power dissipation since the dynamic power is predominant, and the increase in dynamic power for the 0.25 mil channel length is relatively small.

4.1.4 Operating Voltage

The breakdown effects first observed on DD circuits made at RCAL appear to have been due to processing or design rule problems. Some failures have been experienced under test with circuits which have not been screened for 15 volt operation. However, all custom circuits procured for the PWP underwent a 15 volt functional screening test.

The results on the adder circuit indicated that the circuit propagation delay decreases in a range of 15% to 19% as the operating voltage increases from 10 volts to 12 volts. From 12 to 15 volts, the delay improvement is not proportionally as great. These results indicated that 10 MHz operating speed may be obtainable at 12 volts. An operating voltage reduction of 15 to 12 volts has the dual result of better reliability and a power dissipation reduction of 36%.

4.2 FABRICATION PROBLEMS

Two fabrication problems were experienced during the program which impacted on schedule and costs. The first of these was effectively a zero yield situation encountered in the initial attempts by SSTC to fabricate the TCS-057 multiplier. The second problem was time dependent circuit failure which was traced to process contamination and resulted in the requirement that a majority of the circuits fabricated had to be replaced.

4.2.1 Multiplier Circuit Fabrication Yield

The initial design of the TCS-001 9x9 multiplier was developed at RCAL before manufacturing techniques were established at SSTC. In the first attempt to fabricate the TCS-001 at SSTC, a set of mirrored masks were used and although the circuit was functional, a very small yield was obtained. When the replacement (non-mirrored) mask set was used, a zero yield resulted.

Initially, two lots of the TCS-001 were run at SSTC with the deep depletion process. A zero yield was obtained although the same design rules were used on two other chips, a 19 stage sequence generator (TCS-004) and an expandable 8x8 multiplier (TCS-002). Table 10 shows the yield experience for the three chips. The TCS-004 which is about one-fourth the area of the other two chips gave an overall yield of about 10 percent. Based on actual experience (9), this design procedure could be expected to produce a yield of about one percent in the larger multiplier chips. However, as indicated in Figure 14, if the defects are randomly distributed on the chip, a yield of about 0.1 percent may result. No specific problem area on the TCS-001 was identified. However, the established design rules specified a 0.2 mil spacing between epi-islands and polysilicon while the three chips compared here have a 0.1 mil spacing. The fact that the TCS-002 multiplier had a better yield may have been due to a lower incidence of the 0.1 mil spacing condition. Scanning election micrographs (SEM) were

TABLE 10
YIELD RESULTS ON TCS-001

	SIZE OF CHIP	LOTS PROCESSED	CIRCUIT YIELDS
TCS-004	108 x 116 mils	3	15.9%, 2.6%, 14.4%
TCS-002	189 x 195 mils	2	1.15%, 1.54%
TCS-001	210 x 215 mils	2	0%, 0%

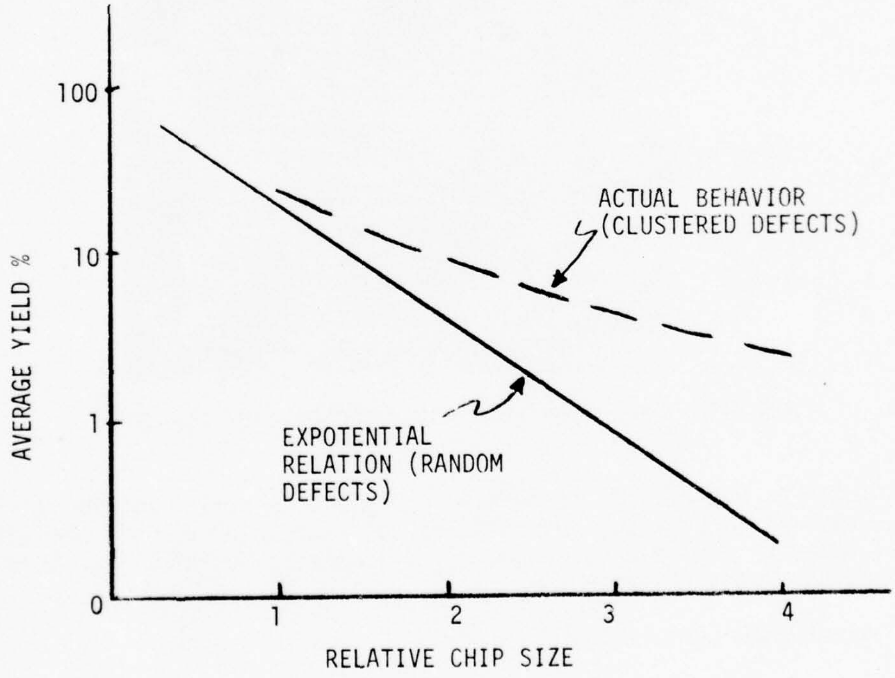


FIGURE 14. YIELD EXPERIENCE AS A FUNCTION OF LSI CHIP SIZE

employed and Figure 15 shows a portion of a TCS-001 chip. The epi-islands and polysilicon touch each other at various locations which could cause an epi-poly electrical short. If this were the case, these latter failures would tend to be randomly located and thus cause the large reduction in yield. The narrow spacing is also a potential cause of the two elements touching and trapping a contaminant. Finally, a narrow spacing also prevents good metal deposition over and in the gap. There is a tendency of partial filling of the gap with metal and a subsequent crack developing thus creating an open circuit condition. This latter effect is illustrated in Figure 15.

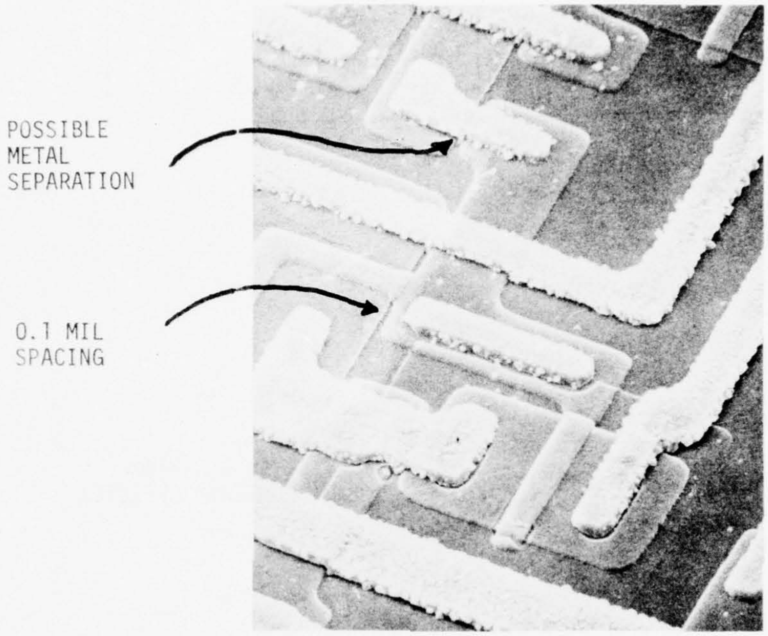


FIGURE 15. SCANNING ELECTRON MICROGRAPH OF TCS-001

An attempt was made at increasing the yield by shrinking the epi-islands in the mask making process, but this was not successful and the decision was made to redesign the chip using the 0.2 mil spacing design rule. Good yields were obtained after the redesign.

4.2.2 Short Term Circuit Instability

4.2.2.1 Problem Background - Unusual difficulty was encountered early in December 1975 during testing of the first sample FFT memory module. It was discovered that some circuits would fail after several minutes to hours of operation. After removal of power for a short time, the circuit would again operate properly and the failure cycle would repeat. In addition, a number of circuits which were judged inoperable at MSRD passed the standard computerized testing at SSTC. It was then found that these latter circuits all failed when the SSTC test was recycled for a few seconds.

The problem has been manifested in other ways. Failures had been experienced on the FFT module after hours of operation which had been attributed to a leakage problem. In addition, during performance tests on TCS-016 and TCS-017 at ATL, some of the circuits from a given date code worked for a few seconds to several minutes before failing. The leakage current on the failing circuits was erratic, starting high and going very low when the chip failed functionally. The circuits always recovered after power was disconnected. It was felt that the leakage specification on the TCS-017 would catch this problem since it would reject circuits with initially high leakage.

Temperature effects were ruled out on the GUA since the power dissipation in the test conditions was very low. Cooling of a failed chip with Freon did not restore operation.

Based on these initial problems, further analysis was conducted on failed chips at SSTC. It was found that the problem was due to a circuit instability which was not detected by the CVBT (Capacitance-Voltage-Bias-Temperature) screening tests in the wafers. A square root of current versus input voltage plot for an inverter pair will show a repeatable pattern in a normal circuit when the bias is swept up to maximum voltage, held for an hour or more and swept back to zero volts as shown in Figure 16. The circuits failing after a period of time produced transfer characteristics and square rooter curves similar to Figure 17.

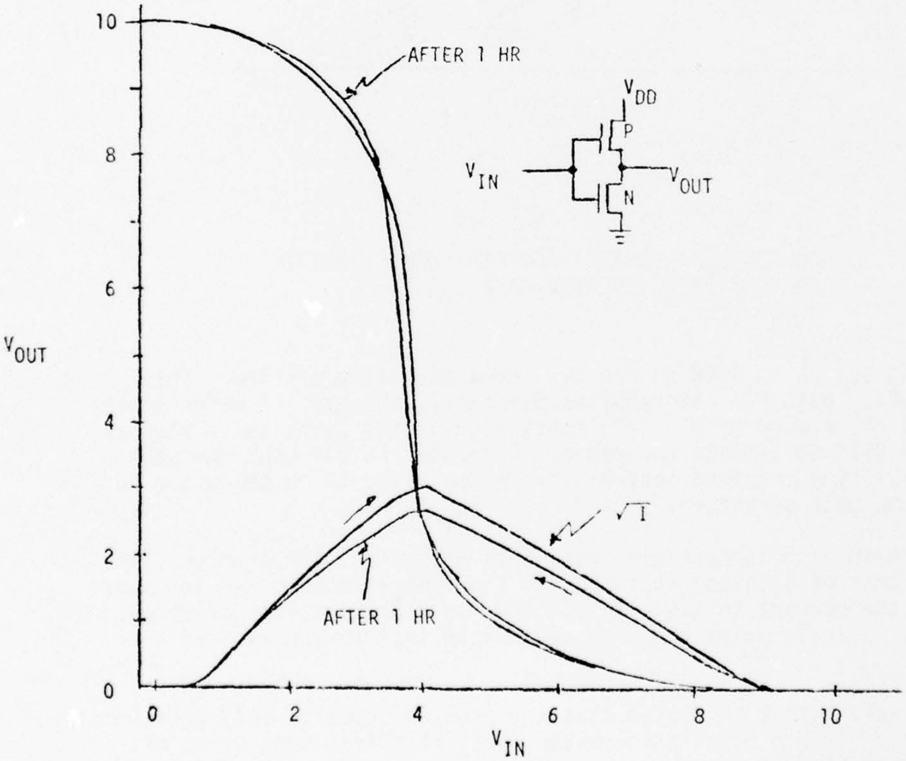


FIGURE 16. SQUARE ROOTER PLOT AND TRANSFER CHARACTERISTIC FOR NORMAL INVERTER PAIR

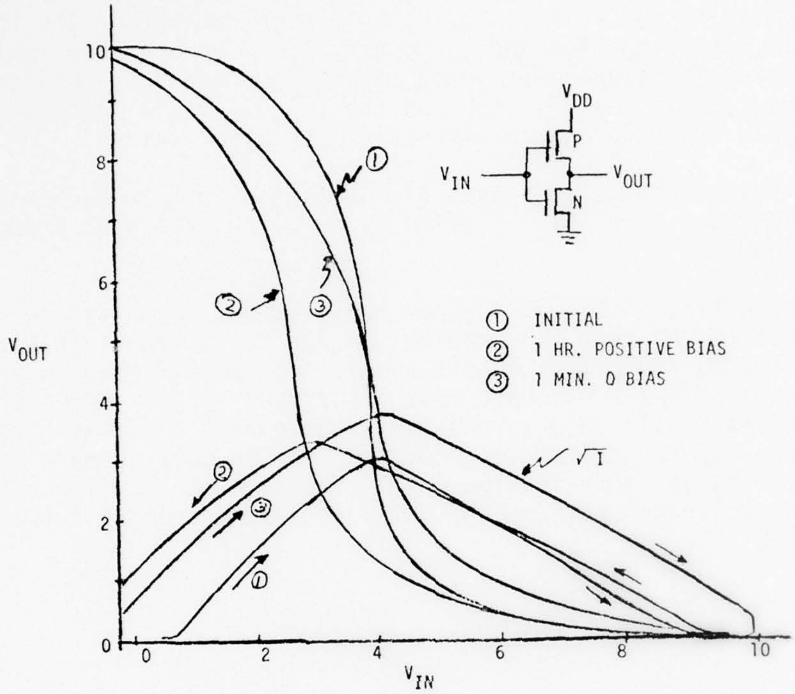


FIGURE 17. SQUARE ROOTER PLOT AND TRANSFER CHARACTERISTIC FOR UNSTABLE INVERTER PAIR

A program was set up at SSTC to resolve the instability problem. This included consultation with RCA Laboratories personnel, analysis of wafer stock to determine time occurrence of problem, contacting device users and a step by step procedure at SSTC to isolate the cause. Personnel at RCA Labs had seen the problem of fast time constant instability and believed it to be unique to boron-doped silicon gate structures.

A normal burn-in high temperature test would not detect the problem. The high temperature test will detect heavy sodium ion contamination, but the short time constant of the current instability problem would prevent its detection. SSTC added a test in their wafer probe procedures to test for short term instabilities.

Analysis of wafer stock indicated that the problem appeared only occasionally and with a severity which gradually increased until the first week or so of November. From that time through December, it was present in virtually every run sampled. It was found to be predominantly wafer dependent. If one chip on a wafer had the problem, no chips have been found on that wafer without the problem.

4.2.2.2 Causes of CMOS Instability - There are four classical causes of instability in CMOS devices.

1. Sodium Ion Contamination - This was ruled out because of its long time constant and the standard screening procedure would detect it.

2. Trapping - Ruled out because of long time constant.

3. Hole Injection - Fast time constant, but mechanism is not known for these devices. Involves avalanche effect, and low operating voltage of these devices rules it out.

4. Proton Motion - This is sometimes caused by "fast sodium" effect and is most likely due to hydrogen ion or proton being injected. This was assumed to be the problem.

The key variables in the fabrication of the devices which were judged to be most likely to cause the instability problem are marked with an asterisk in Figure 18.

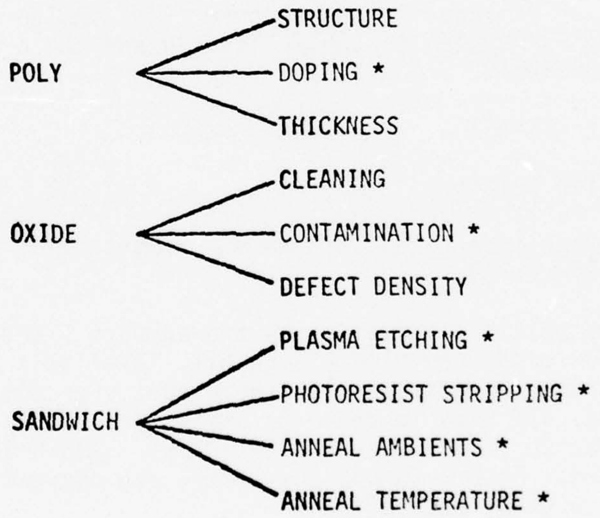


FIGURE 18. CIRCUIT FABRICATION VARIABLES

4.2.2.3 Problem Solution - A procedure was instituted at SSTC to find the source of the problem by a series of tests with runs in line and POS (Poly-Oxide-Silicon) capacitor tests. The POS capacitor tests provided a much quicker procedure than the runs in line since the tests could be made without metal deposition. A total of 192 conditions were possible:

- ° 3 Poly Sources - SSTC is now using its own source of poly as opposed to what SSD uses and its former Poly source.
- ° 2 Clean Options
- ° 2 Photo-Strip Options - SSTC had added a plasma etcher which had greatly improved yield.
- ° 8 Anneal Cycles
- ° 2 Device Types

All 96 conditions were tested with the plasma etcher removed and gave a low incidence of instability. A few runs made with the plasma etcher all produced a high instability rate. With the plasma etcher removed and the standard process re-established, the instability problem has not re-appeared.

4.2.2.4 Problem Effect - Of 714 CMOS/SOS LSI circuits which had been delivered up to the time of the instability problem, only 200 could be identified as definitely being made during a good time frame. It was, therefore, necessary to replace 514 circuits. The total effect of the lost time in finding the problem and replacing the components amounted to a projected schedule slippage of about five months. This delay, and the associated projected additional costs, resulted in a reduction of scope in the program. The implementation goal was reduced from the full PWP system to the FFT's only and the program time frame was also reduced accordingly.

4.3 9 X 9 MULTIPLIER TCS-057

4.3.1 9 x 9 Multiplier Design

The basic sign-magnitude format of the 9 x 9 multiplier designated the TCS-057 is given in Figure 19.

The asynchronous multiplier multiplies two numbers A and B each of which consists of 8 bits for magnitude plus a sign bit. Available at the output are all 16 bits of the product magnitude plus the output sign bit. A roundoff control R is provided. If R is in the high state, the output product magnitude is rounded off to provide an 8 bit plus sign output. The 8 lease significant bits are then not used. With R in the low state, the correct full 16 bit product is available without roundoff.

The asynchronous multiplier contains 64 AND gates which produce the partial products. The partial products are added together using full or half adders to produce the sum. The order of addition is chosen so as to minimize the maximum number of adder stages on any signal path. The first step is to reduce the partial products to two numbers to be added. This is accomplished after four or less additions in any signal path. These two numbers are then added using full adders designed to minimize the carry propagation time. Roundoff, if required, is done during the add so no additional time is lost on the roundoff.

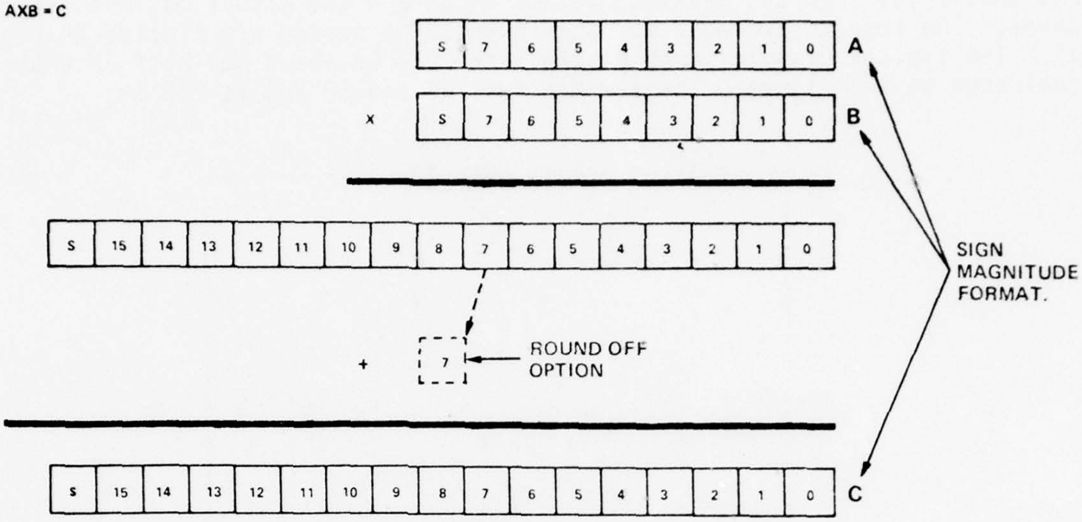


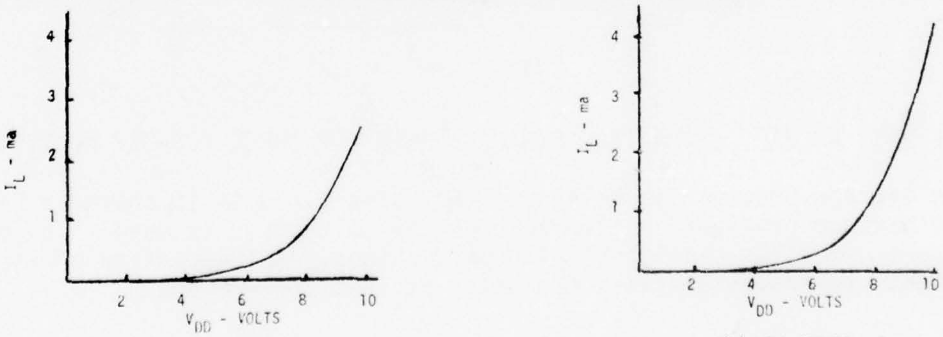
FIGURE 19. 9 X 9 MULTIPLIER FORMAT

The multiplier is expandable. For a 16 bit plus sign multiplier, 4 of the 8 bit plus sign multipliers are needed. In addition, five 8 bit adders plus any retimers required for speed are needed.

4.3.2 Multiplier Performance

Performance measurements were made on three multiplier chips processed I^2N/N with 0.25 mil channel length. These were tested functionally and dynamically. All units passed full functional tests. One of the three circuits which initially failed functional tests at a low voltage was found to be fully operational above V_{DD} = 6 volts.

Leakage current was high for the chips tested as indicated in Figure 20 for chip number 1. The lowest value was 1.0 ma at 10 volts.



(a) ALL INPUTS = 1

(b) ALL INPUTS = 0

FIGURE 20. TCS-057 LEAKAGE TEST FOR ONE SAMPLE

An alternating all ones and all zeros input pattern was used to measure the maximum dynamic power dissipation. The average dynamic power is one-half of the maximum. The leakage current adjusted for the ones and zeros input was subtracted from the measured values to obtain the actual maximum dynamic power. The results averaged for the three chips tested are plotted in Figure 21. The typical dynamic power dissipation will be about one-half of that indicated on the figure. The average for 12V and 10 MHz is 275 mw.

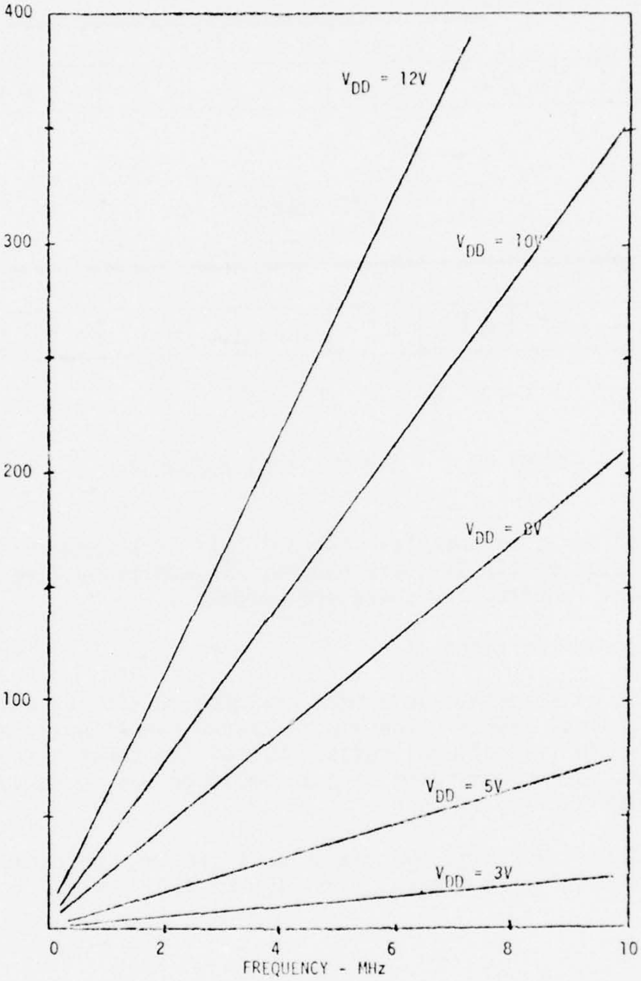


FIGURE 21. AVERAGE OF MAXIMUM DYNAMIC POWER FOR THREE TCS-057 MULTIPLIERS

The average propagation delay for the three circuits is shown in Figure 22. The maximum propagation time through the multiplier is used. The measured results are generally consistent with predictions. Extrapolation of the measurements to 15 volts places the delay at less than 80 nsec.

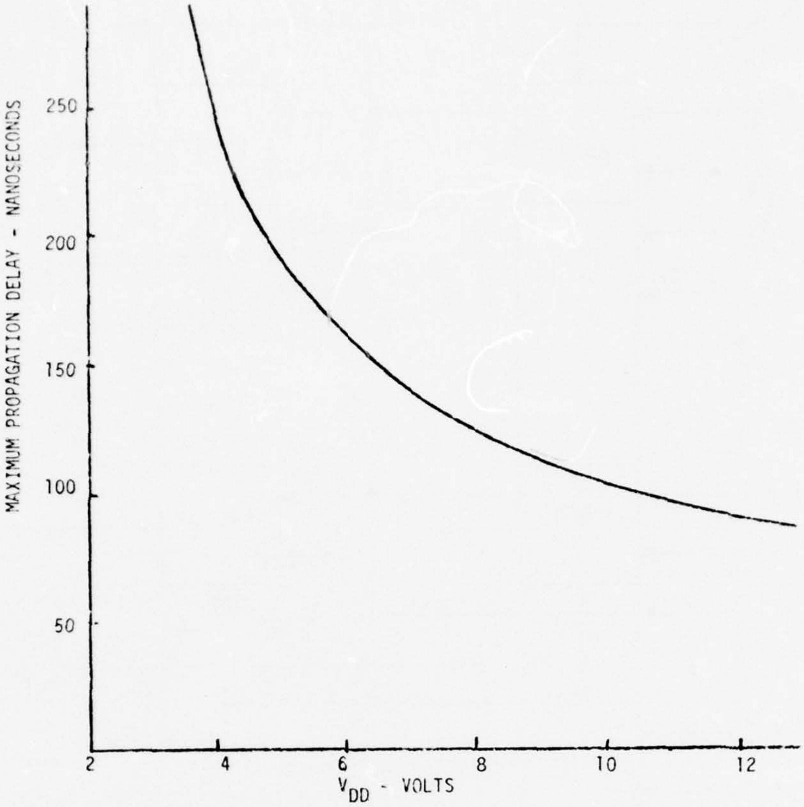


FIGURE 22. AVERAGE PROPAGATION DELAY OF LONGEST PATH FOR THREE TCS-057 MULTIPLIERS

4.4 NINE BIT ADDER TCS-065

4.4.1 9 Bit Adder Design

The adder array TCS-065, shown functionally in Figure 23 adds the two numbers A and B to obtain the sum, C. Each of the inputs consists of 8 bits for magnitude and one bit for sign. A carry in (C_{in}) input is provided which is added into the least significant bit position. Either 1's complement or 2's complement representation for negative numbers can be handled. When used as a 1's complement adder, the end around carry is obtained from the F_{out} output. This end around carry is applied to the C_{in} input. When used as a 2's complement adder, the connection between F_{out} and C_{in} is not made.

When operated in the FFT stage, the output consists of 8 bits for magnitude (C_0, C_1, ..., C_7), a sign bit C_s, and an overflow indication bit O_{out}. The overflow indication bit is in the high state whenever the magnitude of the sum exceeds that which can be represented by 8 bits. Whenever overflow occurs, the output magnitude bits are all shifted one position. The least significant bit

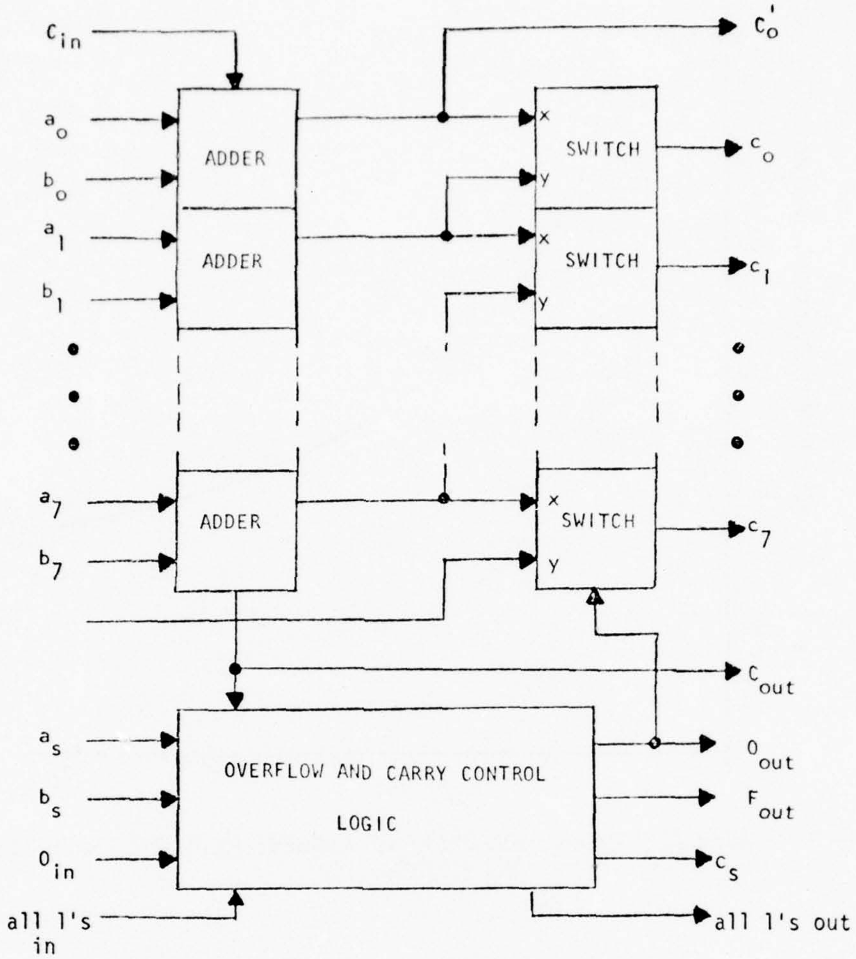


FIGURE 23. 9 BIT ADDER ARRAY, TCS-065

is shifted out, the next least significant bit becomes the least significant bit, etc. The carry out of the last stage of the adder becomes the most significant bit. This requires that the C_{out} output be connected to the y_{in} input. The one bit shift also occurs when the 0_{in} input is in the high state. However, 0_{out} is in the high state only when overflow occurs in the addition. In order to prevent oscillation in the 1's complement mode, an all 1's condition with A and/or B negative is detected which forces an end-around carry-giving a positive-zero output.

The adder is fully expandable. A 16 bit plus sign adder requires two arrays with C_{out}, 0_{in} and y_{in} of the first array connected respectively to C_{in}, 0_{out}, C'_0 of the second array. In addition, for 1's complement C_{in} of the first array is connected to F_{out} of the second array. The first array handles the 8 least significant bits while the second array handles the 8 most significant bits. The sign bits for the two input numbers are applied to a_S and b_S of the second array. Either a_S or b_S in the first array must be high while the other must be low. The overflow indication bit O_{out} of the second array is high and shifting of the output magnitude bits occurs if the sum magnitude exceeds that represented by 16 bits.

The adder array can also be operated to give a full 9 bits of magnitude plus sign bit out with no overflow shifting. This is accomplished by taking the output magnitude bits from C_0', C_0, C_1, C_2, ..., C_7, C_{out} with O_{in} in the high state.

4.4.2 9 Bit Adder Performance

The leakage test results for 10 test TCS-065 adders is given in Table 11. The leakage with some of the test circuits was very high. All of the high leakage circuits were screened out with a leakage test for the PWP. The maximum acceptable leakage was 250 microamp and the vast majority were less than 100 microamp at 10 volts.

TABLE 11
SAMPLE ADDER LEAKAGE TESTS

LEAKAGE CURRENT MICROAMP.	INPUTS LOW		INPUTS HIGH	
	5V	10V	5V	10V
<10	6	5	3	0
10-100	1	0	2	2
100-1000	1	3	1	3
1000-10000	2	1	4	0
>10,000	0	1	0	5

The propagation delay measurements for the 10 test chips are summarized in Figure 24.

4.4.3 TCS-008 Adder Logic Error

During the complex multiplier breadboard tests, a logic error was discovered in the TCS-008 adder array. The error was only evident when the circuit is placed in the floating point format of the PWP FFT arithmetic unit. Specifically, the output of the adder is to be right shifted one bit and truncated when an overflow on either the adder or subtractor circuit occurs. The correct result is obtained in the original TCS-008 adder when the overflow is on the chip, but when it originates on the other chip, an incorrect answer can be placed in the most significant bit position. The error can be corrected by using two exclusive-or circuits external to the adder as shown in Figure 25.

The original intent was to obtain CMOS/SOS quad exclusive-or circuits type 4030 on a selected basis from Inselek Corporation. This would have resulted in a worst case adder propagation delay of about 110 nsec. Furthermore, by eliminating

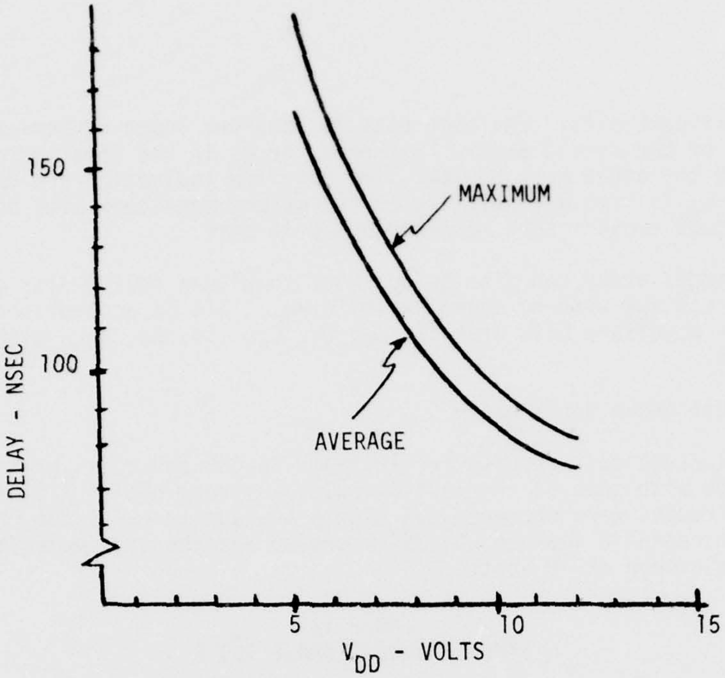
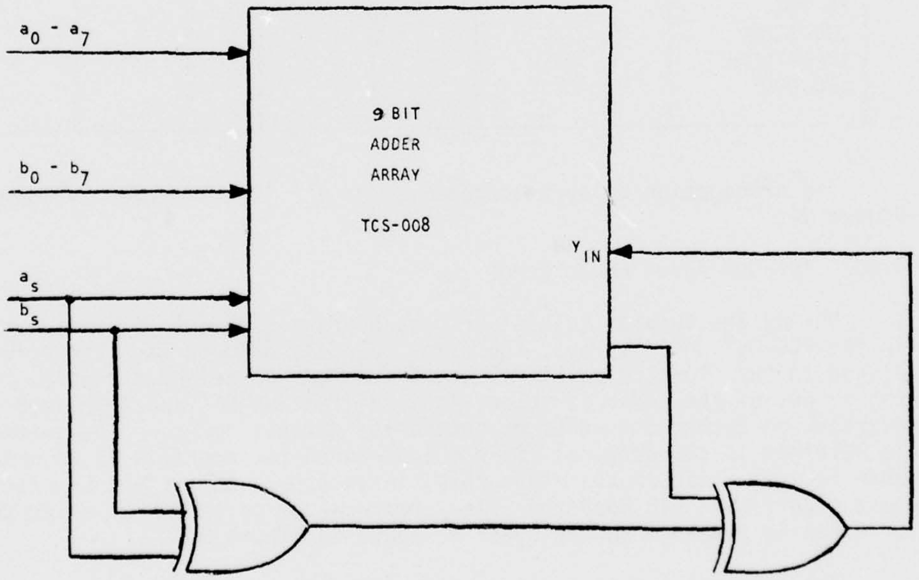


FIGURE 24. PROPAGATION DELAY OF TCS-065 ADDER IN 1'S COMPLEMENT MODE



$Y_{IN} = (a_s \oplus b_s) \oplus C_{OUT}$

FIGURE 25. ADDER CORRECTION CIRCUIT

the end-around carry in the adder and thus taking a small reduction in performance level, the 10 MHz speed could still have been attained. However, early in January 1975, Inselek Corporation filed a bankruptcy petition and it was not possible to procure the CMOS/SOS 4030 parts. The use of bulk CMOS circuits would have provided typical delays of 120 nsec for both the end-around carry and non-end-around carry conditions.

This speed reduction, giving a maximum 8 MHz operating speed for thc PWP, was too great so a redesign of the adder array to correct the error was instituted. The new design is designated TCS-065.

4.5 DUAL FLOATING POINT SCALER, TCS-016

4.5.1 Floating Point Scaler Design

The floating point scaler TCS-016 (see Figure 26) contains two 8 bit shifters. The input to the shifter is an 8 bit word labeled $a_7 \ldots a_0$ where a_0 is the least significant bit and a_7 is the most significant bit. The output is the 8 bit word $b_7 \ldots b_0$ where b_0 is the least significant and b_7 the most significant bit. There are 4 bits in the shift control input which are designated by S_1, S_2, S_4, and S_8 with the subscript indicating the number of places to be shifted. The only other input to the shifter is the sign bit of the input word which is called a_S.

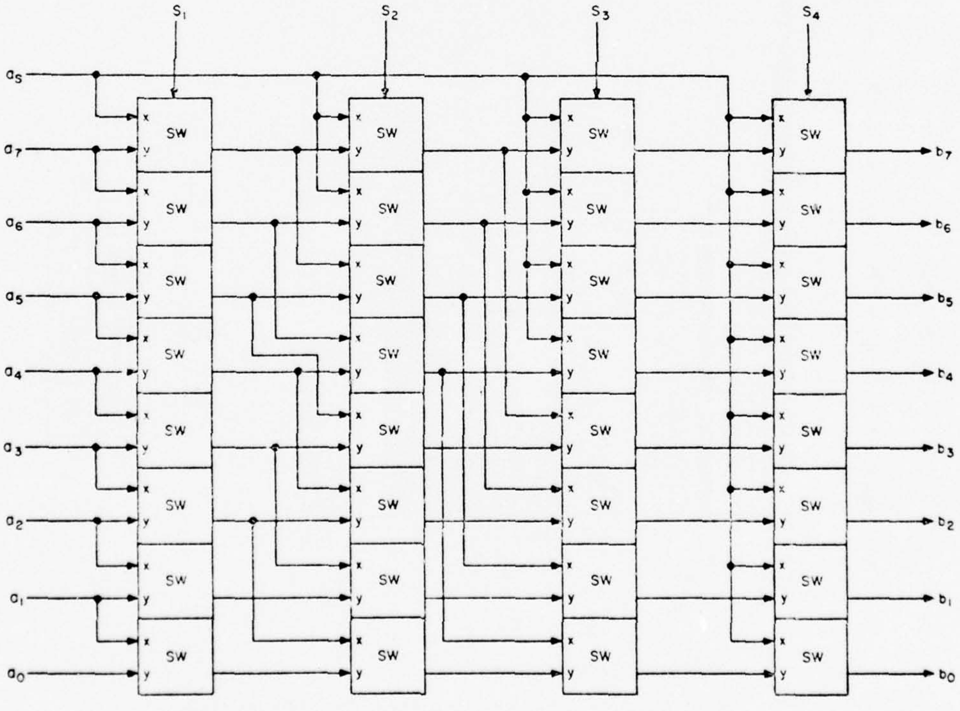


FIGURE 26. FLOATING POINT SCALER (TWO PER ARRAY)

Shifting occurs if any of the shift control inputs are in the high state. In shifting, the least significant bits are lost. The sign bit a_s is shifted into the most significant bits. If S_1, S_2, S_4, and S_8 are all low, the output $b_7 \ldots b_0$ is identical the input $a_7 \ldots a_0$. If only S_1 is high, a shift of 1 place occurs with the output being identical to $a_s a_7 \ldots a_1$. If only S_2 is high the output is identical to $a_s a_s a_7 \ldots a_2$. If S_1 and S_2 are high while S_4 and S_8 are low, a shift of three places occurs with the output becoming $a_s a_s a_s a_7 a_6 a_5 a_4 a_3$. By using various combinations of the shift control inputs any shift between 0 places and 8 places can be obtained. Whenever 8 or greater number of shifts is programmed in, the output is $a_s \ldots a_s$.

4.5.2 Scaler Performance Results

The performance results summarized on the scaler and retimer circuits covers work done under the Manufacturing Methods Contract (F33615-73-C-5043). Dynamic performance tests were made on a total of 36 scaler circuits processed by double-epi with 0.30 mil channel lengths.

4.5.2.1 Leakage Current - The total array leakage current for the 36 scaler arrays was measured under two conditions. First, the current was measured with all inputs at ground and V_{dd} applied to the chip. Secondly, the leakage current was measured with all the inputs at V_{dd}. The readings under the two conditions often differed greatly, but neither was consistently greater than the other. The leakage current distribution at 5, 10, and 12 volts is shown in Figure 27.

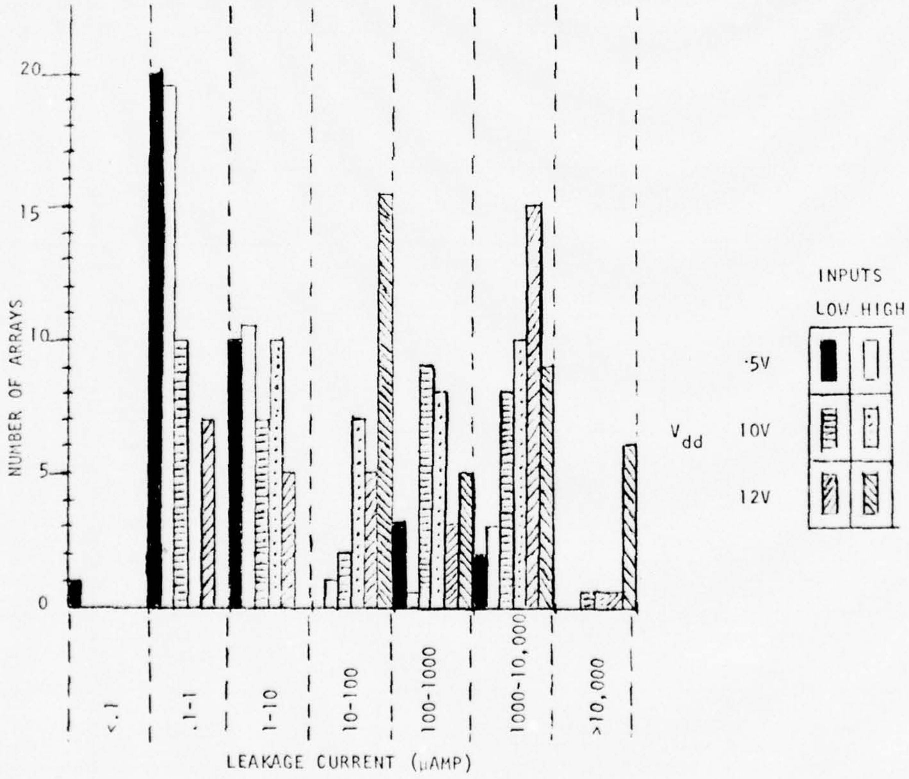


FIGURE 27. DUAL 8-BIT SCALER LEAKAGE DISTRIBUTION

4.5.2.2 Propagation Delay - The Floating Point Scaler array consists entirely of combinatorial logic which makes the propagation delay between inputs and outputs the limiting factor in speed of operation. The propagation delay was measured for several different paths through the array. The first path considered is between the bits of the 8-bit input word and bits of the 8-bit output word. The signal for this path must pass through four switches plus the output driver. At a V_{dd} of 10 volts, this delay was measured between each input bit and all possible output bits. Little variation was observed within an array. Therefore, only the delay between the MSB input and each bit of the output word was measured. The results are given in Table 12. The delays are slightly greater when the output makes a negative transition than they are when the output makes a positive transition. The output load is approximately 5 pf. Figure 28 shows the delay as a function of V_{dd} voltage.

TABLE 12
INPUT WORD a TO OUTPUT WORD b PROPAGATION DELAY FOR FLOATING POINT SCALER ARRAY

V_{dd}	POSITIVE TRANSITION	NEGATIVE TRANSITION
5 Volts	87 ns	77 ns
7 Volts	55 ns	50 ns
10 Volts	38 ns	34 ns
12 Volts	32 ns	30 ns
15 Volts	27 ns	25 ns

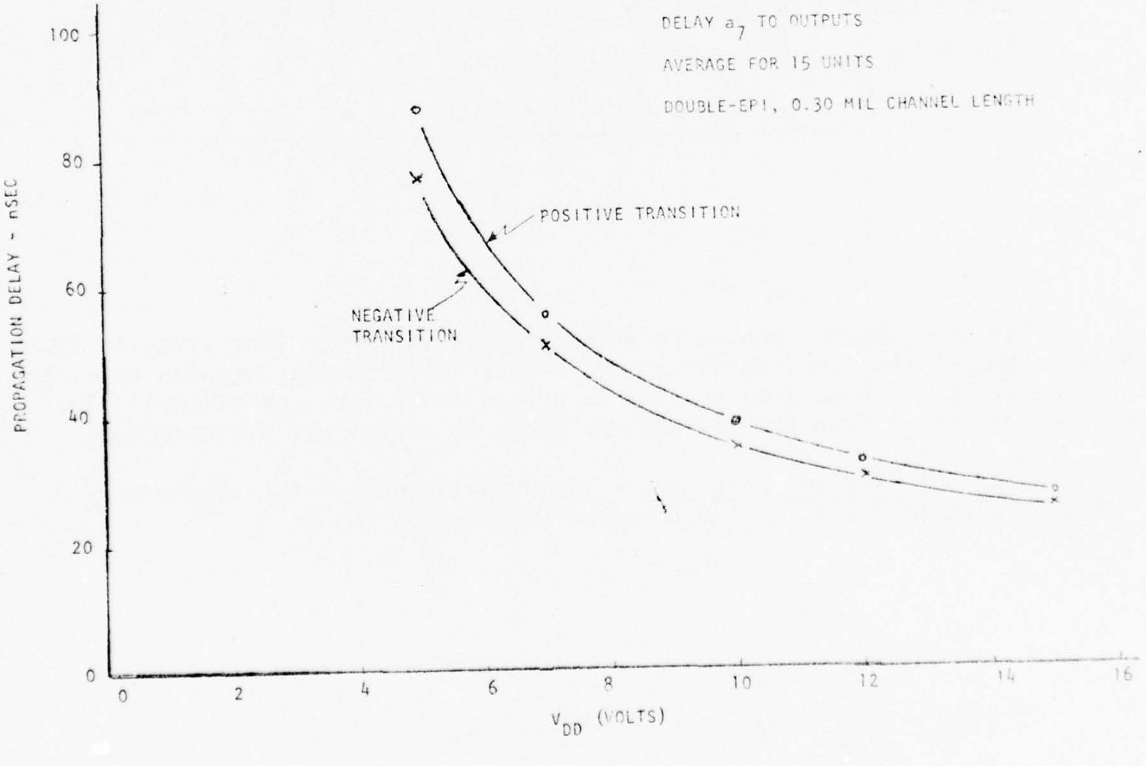


FIGURE 28. SCALER DELAY VERSUS OPERATING VOLTAGE

The longest propagation path in the array is from the shift-1 control input to the bits of the output word. In this signal path are the two inverters of the shift driver, four switches, and the output driver. The average delay is given in Table 13 for operating voltages of 5, 10, and 15 volts. The predicted delay of 23 nanoseconds was based upon the path from shift-1 input control to output with a V_{dd} of 15 volts and channel length of .25 mils. The larger values observed can be attributed in part to the use of the 0.3 mil channel length mask for the test chips instead of the faster 0.25 mil channel length.

TABLE 13
FLOATING POINT SCALER DELAY BETWEEN SHIFT INPUTS AND OUTPUT

	V_{dd} = 5V		V_{dd} = 10V		V_{dd} = 15V	
	POS. TRANS.	NEG. TRANS.	POS. TRANS.	NEG. TRANS.	POS. TRANS.	NEG. TRANS.
SH1 to Output	100 ns	95 ns	48 ns	47 ns	32 ns	32 ns
SH2 to Output	84 ns	71 ns	40 ns	37 ns	28 ns	26 ns
SH4 to Output	60 ns	60 ns	27 ns	28 ns	20 ns	21 ns
SH8 to Output	45 ns	42 ns	22 ns	21 ns	17 ns	16 ns

Delay measurements showed relatively small variation from array to array. As an example, the standard deviation for the 10 volt delay between the input and output is 2 nanoseconds for the 33 arrays which were operational. The maximum deviation from the average for these 33 arrays was 5 nanoseconds.

4.5.2.3 Output Rise and Fall Time - The output rise and fall times vary linearly with load and are indicated in Table 14.

TABLE 14
SCALER RISE AND FALL TIMES VERSUS LOAD

V_{dd}	LOAD			
	5 pf		20 pf	
	RISE TIME	FALL TIME	RISE TIME	FALL TIME
10V	10.7	9.5	17.5	14.0
15V	9.0	8.5	13.0	11.5

4.5.2.4 Power - Dynamic power dissipation was measured under a variety of conditions. Three cases were considered.

Case 1: Inputs - In phase with adjacent inputs 180° out of phase
Shift Controls - All at ground
(Maximum Power Condition)

Case 2: Inputs - In phase with adjacent inputs in phase or 180° out of phase
Shift Controls - Shift - 8 high, other shift controls varying
(Internal dissipation of first three switch positions)

Case 3: Inputs - Ground
Shift Controls - Varying
(Measures dissipation of shift drivers)

An estimate of the average dynamic power dissipation can be made by assuming that each input and output signal has a 50% probability of changing with each new input word. A good approximation to this is the average of cases 1 and 3. This is included in Table 15 with the results of the individual cases. These dynamic power results are greater than the predicted values for a 5 pf load of 5.7 mW/MHz at 10 volts and 12.9 mW/MHz at 15 volts. The 0.30 mil gate lengths implemented rather than the 0.25 mil assumed may account for some of the difference.

TABLE 15
DYNAMIC POWER DISSIPATION OF SCALER

V_{dd} (VOLTS)	DYNAMIC POWER DISSIPATION (mW/MHz)			
	Case 1	Case 2	Case 3	Average
5	2.5	0.7	0.8	1.6
7	5.4	1.5	1.6	3.5
10	11.8	3.4	3.4	7.6
12	17.9	5.1	5.2	12.0
15	30.3	8.9	9.2	19.7

4.5.2.5 Temperature - The propagation delay of five scaler arrays was measured during temperature cycling from ambient to 125°C. No functional failures occurred during the tests. The average delay increases in a close to linear fashion about 25% as the temperature varies from 25°C to 125°C.

4.6 RETIMER REGISTER, TCS-015

4.6.1 Retimer Register Design

The retimer register TCS-015, shown functionally in Figure 29 contains two sets of register circuits each of which is capable of retiming 9 input bits. In addition to retiming the input, the register can also be used to complement input signals. When control input H is in the low state, outputs C_0 through C_7 are the same as inputs a_0 through a_7 except for the one clock period delay of the retimer. When H is in the high state, the outputs C_0 through C_7 are the complements of the inputs a_0 through a_7 except for the one clock period delay. For the 9th input bit a_s, both a complemented and an uncomplemented output is provided. In the FFT stage the a_s input would be the sign bit for the input number being retimed.

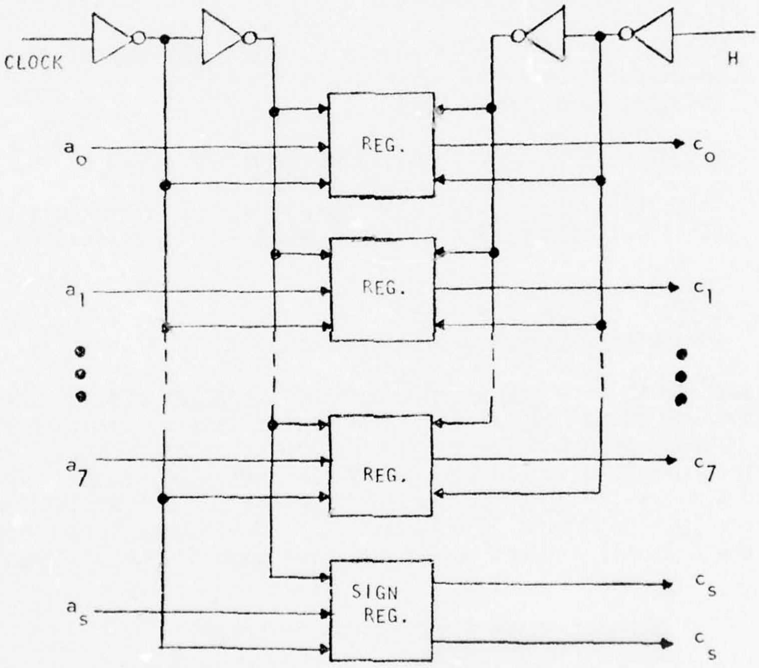


FIGURE 29. RETIMER REGISTERS (2 PER LSI ARRAY)

Clock is generated on the chip from the clock input eliminating the need for both a clock and a $\overline{\text{clock}}$ input.

Static registers are used which eliminates a low limit on speed of operation.

The two register circuits on the array are independent of each other with each having its own clock and H inputs.

4.6.2 Retimer Register Performance

Extensive tests were made on 20 Retimer register arrays. Five were processed with the I^2N/N (ion-implantation) process, four by deep depletion and eleven by the standard double-epi process. All had 0.3 mil channel lengths.

4.6.2.1 Leakage Current - Large variations in the leakage current were observed for the ion implantation and deep depletion arrays. Measurements at 10 volts produced leakage values from 4 μa to over 10 ma for the nine arrays in these groups.

The eleven double-epi arrays showed little variation from array to array or between the inverting and non-inverting conditions. The range of values observed is shown in Table 16.

TABLE 16
DOUBLE-EPI RETIMER REGISTER LEAKAGE MEASUREMENTS

V_{dd} (VOLTS)	LEAKAGE CURRENT (μAMP)		
	MEDIAN	MAXIMUM	MINIMUM
5	2	30	1
7	4	35	2
10	21	48	3
12	41	200	22
15	130	360	100

4.6.2.2 Speed - A comparative measure of the propagation delays of the retimers was made by connecting them in an 18 bit shift register configuration. No noticeable difference in speed was observed due to the different fabrication processes. However, if the complementing control (H) was set, the speed was reduced since this adds a gate to the signal path. In addition, with the test pattern of alternate "1's" and "0's" being used, all input and output data is in phase when H is "1", while the data input and output of adjacent stages is 180° out of phase when H is "0". This results in increased capacitive loading in the test fixture when H is "0" which reduces the maximum speed. The average maximum operating speed for the register is given in Table 17.

4.6.2.3 Output Delay, Rise and Fall Times - The delay is measured between the 50% point of the positive transition of the input clock and the 50% point of the

TABLE 17
MAXIMUM RETIMER FREQUENCY WHEN OPERATED AS SHIFT REGISTER

V_{dd}	Maximum Frequency (MHz)					
	I^2 (N/N)		Deep Depletion		Double-Epi	
(Volts)	H = 0	H = 1	H = 0	H = 1	H = 0	H = 1
5	17.0	21.2	17.1	20.9	17.9	22.2
7	23.2	30.1	24.4	29.2	23.7	29.1
10	30.4	40.2	30.5	38.7	30.3	37.4

resultant output data transition. This delay includes the delay of the clock drivers, the turn-on time of the register output transmission gate, and the delay in the output drivers. The output load consists of the package, test fixture, and scope-probe capacitance plus additional capacitance which may be added.

Delay results for the double-epi retimer arrays are shown in Table 18 and Figure 30. The difference in delay between the OS (sign) and $\overline{OS}$ outputs results from the additional inverter in the OS output. These delays are the same as those predicted by computer simulation. The delay results for the deep depletion and ion implantation arrays are slightly smaller than the double-epi values.

TABLE 18
CLOCK TO OUTPUT DELAY FOR DOUBLE-EPI RETIMER ARRAYS

	V_{dd}	Clock to Output Delay (ns)			
Output	(Volts)	Load = 5 pf	10 pf	15 pf	20 pf
DATA	5	31	34	37	39
OS	5	42	45	48	50
$\overline{OS}$	5	35	39	42	44
DATA	10	18	20	21	23
OS	10	25	26	27	28
$\overline{OS}$	10	22	24	25	26
DATA	15	14	16	17	18
OS	15	19	21	22	22
$\overline{OS}$	15	16	18	20	21

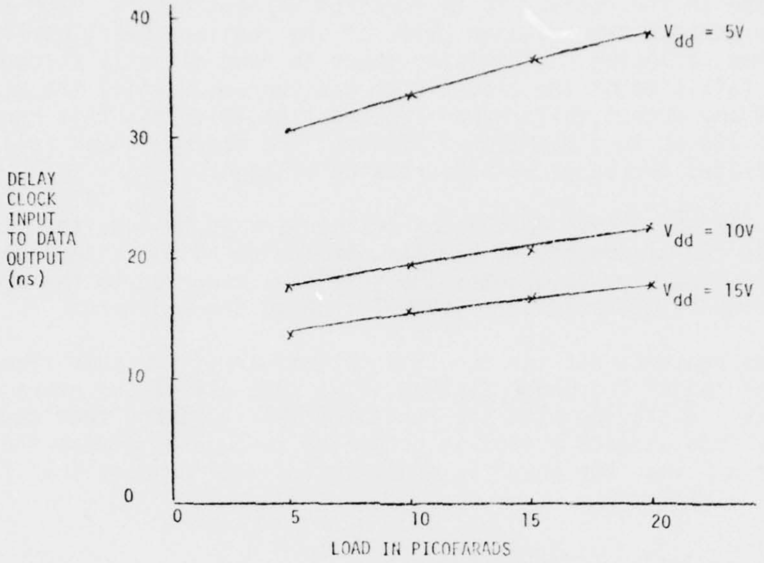


FIGURE 30. DELAY VERSUS LOAD FOR DOUBLE-EPI RETIMER ARRAY

The output rise and fall times are measured between the 10% and 90% points of the data transition. The predicted rise and fall times are 18 ns for a 15 pf load and 11 ns for a 5 pf load when operating at 10 volts. The actual observed values are less than these predicted values. Little difference is observed between arrays made by the different processes. Rise and fall times for the data outputs of the double-epi arrays are given in Table 19.

TABLE 19
RETIMER DATA OUTPUT RISE AND FALL TIMES

V_{dd} (Volts)	Load pf	Rise Time ns	Fall Time ns
5	5	16	12
	10	22	17
	15	30	20
	20	38	23
10	5	11	10
	10	14	12
	15	16	13
	20	18	15
15	5	10	8
	10	12	10
	15	13	11
	20	15	13

The input clock to the retimer array must have a fall time not exceeding some critical value if the retimer is to function without errors. Too slow a fall time results in both transmission gates of the register being partially on at the same time, allowing the register input to feed directly through to the output. The fall time of the clock input was increased until the maximum fall time without any output failure was reached. At 10 volts, this ranged from a minimum of 160 ns to a maximum of 270 ns. The maximum clock fall time which can be tolerated decreases with increasing voltage.

4.6.2.4 Power - For the eleven double-epi retimer arrays tested, the static power is less than 150 microwatts at 5 volts, 480 uW at 10 volts, and 540 uW at 15 volts. This static power becomes insignificant compared to the dynamic power when the array is operated at frequencies above one megahertz.

Dynamic power measurements for the full retimer array are summarized in Table 20. A breakdown of the power figures shows that 42% of the average power results from clock transitions with the remaining 58% resulting from data transitions. The load on each output is estimated to be 5 pf during the power measurements. With H low, the power is approximately 10% greater than it is with H high.

TABLE 20
AVERAGE RETIMER DYNAMIC POWER

V_{dd} (Volts)	Power (mW/MHz)
5	2.9
7	5.8
10	12.8
15	32.4

The dynamic power for the deep depletion and ion implantation arrays was approximately 20% lower.

4.6.2.5 Temperature - The deep depletion and ion implantation retimer arrays were tested at a temperature of 100°C in addition to room temperature. Seven of nine arrays continued to function correctly at the higher temperature, but with increased power dissipation and decreased speed. The increased power results from an increase in both the leakage current and the dynamic current. The observed dynamic power increase was 12% when the array was operated at 5 volts and 16% when operated at 10 volts.

Four of the double-epi arrays were tested at a temperature of 125°C without any functional failure. The average dynamic power increased by approximately 6% when the ambient temperature was increased from 25°C to 125°C.

4.7 FLOATING POINT LOGIC ARRAY, TCS-017

4.7.1 Floating Point Logic Array Design

The two main functions of the floating point logic array TCS-017 shown in Figure 31 are to determine the exponent in floating point representation and to provide the control signals for the floating point scaler arrays.

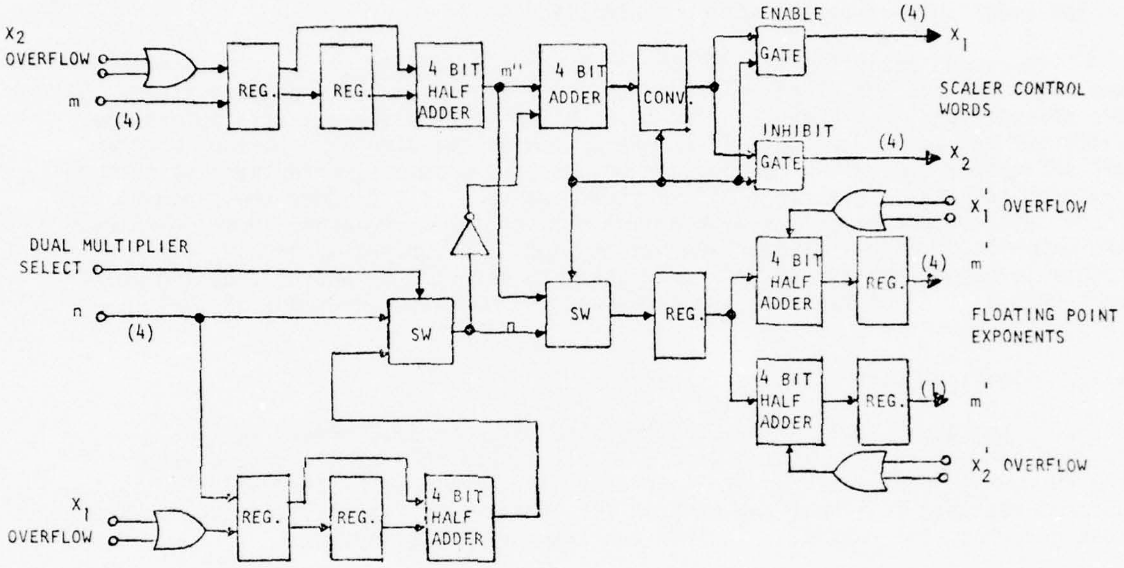


FIGURE 31. FLOATING POINT LOGIC ARRAY

A four bit half adder is used to increment m by one if either of the two multiplier overflow bits is in the high state. The result is labeled m". To align the X_3 overflow bits and the X_1 exponent correctly in time, the exponent is passed through two retiming registers and the overflow bits through one retiming register.

A provision is also made for the use of a multiplier on the input word X_1. This function would be used if a higher order radix FFT were implemented in order to achieve a higher processing rate.

The exponent n is subtracted from m" to determine if either of the X_1 or X_2 numbers must be shifted and by how much before they can be added. The subtraction is accomplished by passing n through an inverter to obtain a 1's complement form which is then added to m'' using a 4 bit full adder with end around carry. If there is a carry out of the most significant bit (MSB) position, the result is assumed to be positive. If there is no carry out of the MSB position, the result is assumed to be negative and in 1's complement form. The output of the adder is passed through a converter which inverts it if there was no carry out of the MSB position of the adder. The output of the converter is ANDed with the carry out of the MSB of the adder to produce the scaler control for X_1, and is ANDed with the not of the carry out of the MSB of the adder to produce the scaler control for X_2.

The second basic function of the floating point logic array is determining the exponents for the floating point representation of the FFT outputs x_1' and x_2'. The components m'' and n are applied to a switch which selects as its output the larger of the two. The control for the switch is the sign bit obtained when n was subtracted from m''. The four bit output of the switch, after passing through retiming registers, goes to a pair of half adders. In one adder the exponent is incremented by one if either X_1' overflow bit is high. The other adder increments the exponent if either X_2' overflow bit is high. The output of each of these adders is passed through retiming registers to give the m' and n' floating point exponents at the output at the same time as the magnitude portions of their respective numbers.

4.7.2 Floating Point Logic Performance

4.7.2.1 Leakage Current - The first measurement on the 20 arrays was for leakage current. This was measured with all inputs held at ground and with the X3 and X4 inputs at V_{dd}. In each case, the array was clocked prior to taking current readings to remove any unknown states from the internal registers. The leakage current measurements include any leakage on the inputs.

The leakage current distribution for 13 of the arrays is shown in Figure 32. The leakage current for the other 7 arrays did not remain constant but drifted downward by as much as an order of magnitude over a period of minutes after power was applied. Consequently, no leakage current readings were recorded for these arrays. All 7 of these arrays were processed in a common batch. The leakage current for some arrays showed considerable variation between the two test conditions. For example, at 10 volts, array 3 had a leakage of 31 μa with inputs low and 3.7 ma with inputs high. At the same voltage, the leakage for array 17 is 2.4 ma with inputs low and 47 μa with the inputs high.

4.7.2.2 Power Measurement and Functional Tests - The operation of the Floating Point Logic Array was checked and power dissipation as a function of clock frequency was measured under two test conditions. In each test, the four X3 inputs receive a pattern consisting of alternating 1's and 0's. In test 1, the four X4 inputs receive the complement of the test pattern on the X3 inputs, while in test 2, the X3 and X4 inputs are the same. All other inputs in both tests are low except for the Dual Mode Select input which is high. Test 1 gives alternating patterns on the SX3 and SX4 outputs with the X5 and X6 outputs remaining high. Test 2 gives alternating patterns on the X5 and X6 outputs while the SX3 and SX4 outputs remain low. By changing a bit in the input pattern, the proper delay through the array can be verified.

Table 21 gives the average dynamic power as determined by the slope of the power versus frequency curve.

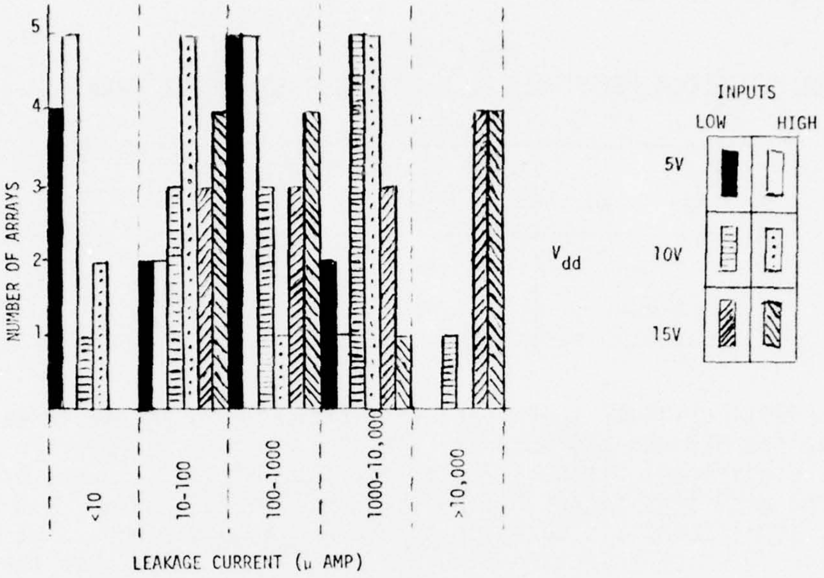


FIGURE 32. INITIAL LEAKAGE CURRENT DISTRIBUTION IN FLOATING POINT LOGIC ARRAY

TABLE 21
FLOATING POINT LOGIC ARRAY DYNAMIC POWER

V_{dd} (Volts)	Dynamic Power (mw/MHz)	
	Test 1	Test 2
5	2.5	3.4
10	13	16
15	33	41

The output load consists of the package and test fixture capacitance which was estimated to be about 5 pf. Since the X3 and X4 inputs are changing at the maximum rate, these power figures are higher than the levels to be expected when the array is operated in the PWP system. Also, if the channel length is reduced from .3 mils to .25 mils, the corresponding reduction in gate capacitance will reduce the dynamic power dissipation. Taking these factors into consideration, these dynamic power results are consistent with the predicted five pf load dissipations of 7.9 mw/MHz at 10 volts and 17.8 mw/MHz at 15 volts.

4.7.2.3 Speed Tests - The maximum clock rate at which the array would operate was measured at 5 volts and 10 volts for the two test patterns used in the functional and power tests. The maximum frequency was not measured at 15 volts since the power dissipation would exceed 1 watt. The average maximum obtainable frequency is given in Table 22.

TABLE 22
MAXIMUM CLOCK FREQUENCY OF FLOATING POINT LOGIC ARRAY

V_{dd} (Volts)	Maximum Frequency (MHz)	
	Test 1	Test 2
5	13	20
10	20	28

A potentially critical speed path within the FFT stage includes the delay from clock to the SX3 and SX4 outputs. This occurs when a carry is generated in the least significant bit position of the four bit full adder, propagates through to the most significant bit position of the full adder, and then back to the least significant bit position by the end around carry. The output selection network which selects either SX3 or SX4 for output has the end around carry as its input. This delay was measured from the retimer of the X3 input and the retimer of the X4 input. The additional transmission gate switch and inverter in the path from the X4 retimer makes that path slightly longer. Table 23 contains the results of this delay measurement. The measured delay includes the delay in the on chip clock driver circuit.

TABLE 23
MAXIMUM DELAY FROM CLOCK TO SX3 OR SX4 OUTPUT

Array Number	X3 Retimer to Output			X4 Retimer to Output		
	V_{dd} = 5V	V_{dd} = 10V	V_{dd} = 15V	V_{dd} = 5V	V_{dd} = 10V	V_{dd} = 15V
2	152 nsec.	74 nsec.	58 nsec.	168 nsec.	83 nsec.	65 nsec.
3	121	66	53	130	74	59
4	159	77	58	166	84	63
5	91	50	40	96	56	45
8	95	52	41	100	58	46
11	99	54	42	110	61	48
13	78	44	36	84	49	40
14	80	44	35	84	49	40
15	124	61	47	134	67	50
19	100	51	39	108	56	44

When the gate lengths are reduced to .25 mils, these delays decrease. The predicted results from the simulations are 40 nsec. for the path from the X3 retimer and 45 nsec. for the path from the X4 retimer when operated at 15 volts. The results are in general agreement with the predicted delay.

4.8 GATE UNIVERSAL ARRAY (GUA)

4.8.1 Gate Universal Array (GUA) Design

A CMOS/SOS Gate Universal Array programmable shift register was developed for the PWP memories. The GUA design approach is one in which only the final metalization layer is defined by the circuit designer. This connects a standard logic gate layout to perform the specific function required. The advantages of this approach are that the design time, fabrication time and costs are reduced. This technique has been well established for bulk CMOS devices. The GUA for the PWP was the first GUA design using CMOS/SOS technology. In the initial fabrication of the circuit, the standard bulk CMOS GUA array was employed. The SOS circuit gave an expected 2:1 improvement in speed over the bulk CMOS GUA. A new design for the CMOS/SOS GUA was developed which:

1. Eliminated the power supply bus which had added excess internal capacitance.

2. Decreased the channel length from 0.3 to 0.25 mils.

3. Used silicon gate processing.

The propagation delays for the bulk CMOS, bulk design implemented in SOS and the improved CMOS/SOS design are given in Figure 33. These measurements were made with a standard GUA test cell.

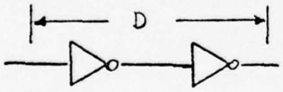
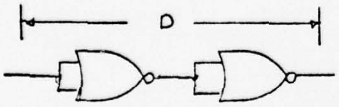
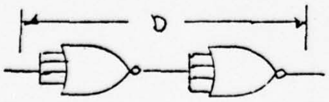
	BULK GUA	BULK DESIGN WITH SOS GUA	NEW SOS GUA
	14	7.5	3.6
	22	19.4	12.7
	66	31.3	27.9

FIGURE 33. GUA COMPARATIVE PERFORMANCE LEVELS OF TEST CIRCUIT

4.8.1.1 GUA Design Procedure - In the CMOS/SOS process, seven mask levels are required for each design. In the GUA, six of the seven mask levels are fixed for each array size. It is the seventh mask level, the metal mask, that is unique for each custom design.

The six mask levels define P-MOS devices, N-MOS devices, P^+ and N^+ tunnels, zener diodes, and pads in a fixed pattern. All drains, sources, gates, tunnel ends, and pads are accessible for interconnection by the metal mask level.

The custom design for the GUA begins with a standard logic design. Standard logic is then transformed into GUA logic cells. A preliminary logic cell placement is done, and the adhesive-backed drawings of the logic cells are then placed on a Mylar sheet. The logic cells are connected by pencil lines that represent the metal interconnection. Standard forms for defining logic cell placement, pad and pin connections, signal types and levels, and test pattern sequences are provided to simplify documentation.

4.8.1.2 Programmable Shift Register GUA Design - The PWP GUA design (Figure 34) incorporates all of the functions necessary for the Input Buffer, Output Buffer, and FFT Memory.(2) This is accomplished by making the shift registers variable length, providing input and output switching, making the shift register clocks independent, incorporating one's complement to sign-magnitude converters (exclusive-or) in both output lines and by control to adjust the clocked delays.

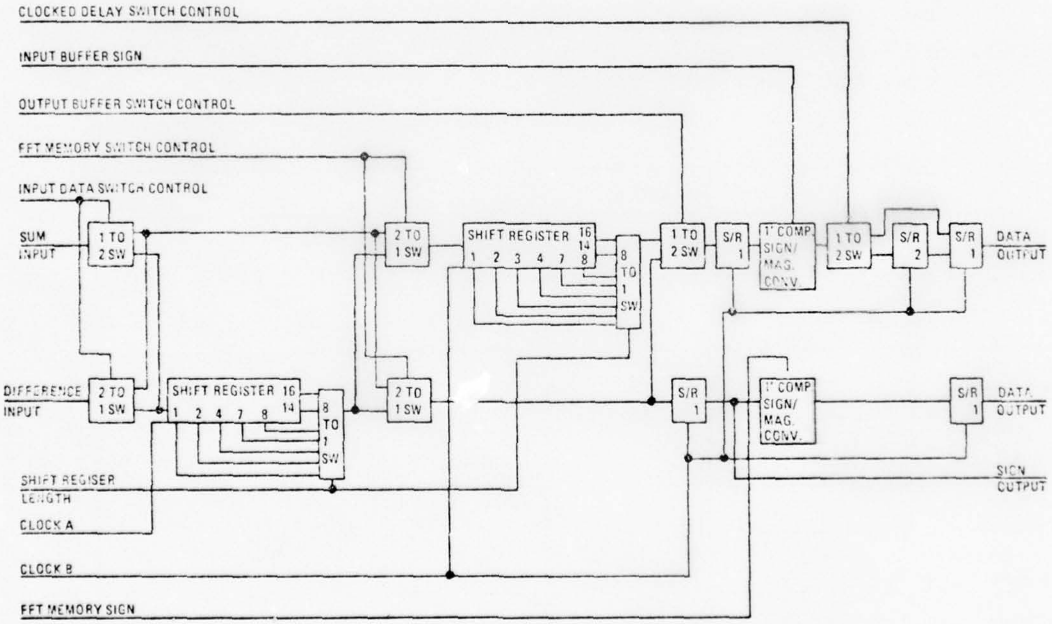


FIGURE 34. PROGRAMMABLE SHIFT REGISTER GUA DESIGN

The shift registers and the clocked delays are dynamic shift register logic cells. These cells require complemented two-phase clocks. The two-phase clocks are generated external to the Universal Array by the control system. The two-phase clock complements are generated on the Universal Array. The dynamic shift register logic cell was used because it occupies only half the number of internal cells as the static shift register logic cell. The dynamic shift register logic cell is constructed from inverters and transmission gates. The clock uses inverters to turn on and off the transmission gates to pass the data.

The two-to-one and one-to-two switches are comprised of two transmission gates and an inverter for controlling the transmission gates. As CMOS transmission gates pass information in either direction, they can be used as either a two-to-one or one-to-two switch.

The shift register length is controlled by a three-bit binary code which is decoded on the GUA. The decoded signal is used to control eight transmission gates. The eight transmission gate outputs are tied together to form an eight-to-one switch.

The one's complement to sign-magnitude converters are two input exclusive-OR logic cells.

All inputs and outputs are buffered through the appropriate devices and are available at the bonding pads which total 20 for the device.

4.8.2 GUA Performance Results

Delays in the PWP GUA fabrication cycle were experienced due to errors in the metalization pattern. However, the re-cycling times for correction were relatively short pointing up the advantages to the GUA approach.

Samples of the PWP GUA (TCS-060-400b) were given functional, leakage and dynamic tests. The units were operated with a 15 pf load at shift register clock rates up to 17.9 MHz. This rate was limited in part by the instrumentation. Typical worst case propagation delay was 40 nsec at 10V. The units tested has high leakage, but this was not unexpected since the test samples had not undergone normal screening procedures after fabrication. All devices fabricated for use in the system had a maximum of 250 μa leakage.

4.8.3 GUA Dynamic Shift Registers

A highly variable yield was experienced during the quantity fabrication of the TCS-060-400b. Extensive studies were made to try to ascertain the cause but no definitive problem was identified. However, it is felt to be due to the dynamic shift register design of the memories. Until a detailed analysis can isolate the problem, the dynamic register is not recommended for further implementation with the CMOS/SOS GUA.

In retrospect, the design of dynamic shift register memories has a large impact on system hardware performance and simplicity. A two-phase clock signal is required by the dynamic shift registers which complicates system timing. In addition, shift register memories require clocking of all memory stages at every clock pulse. Thus, the total system clock load and power dissipation is high.

A preferred choice for future designs would be to use random access memories as the basic memory store. This would greatly reduce clock driver requirements and dynamic power consumption for a given memory size.

4.9 CMOS/SOS RAM

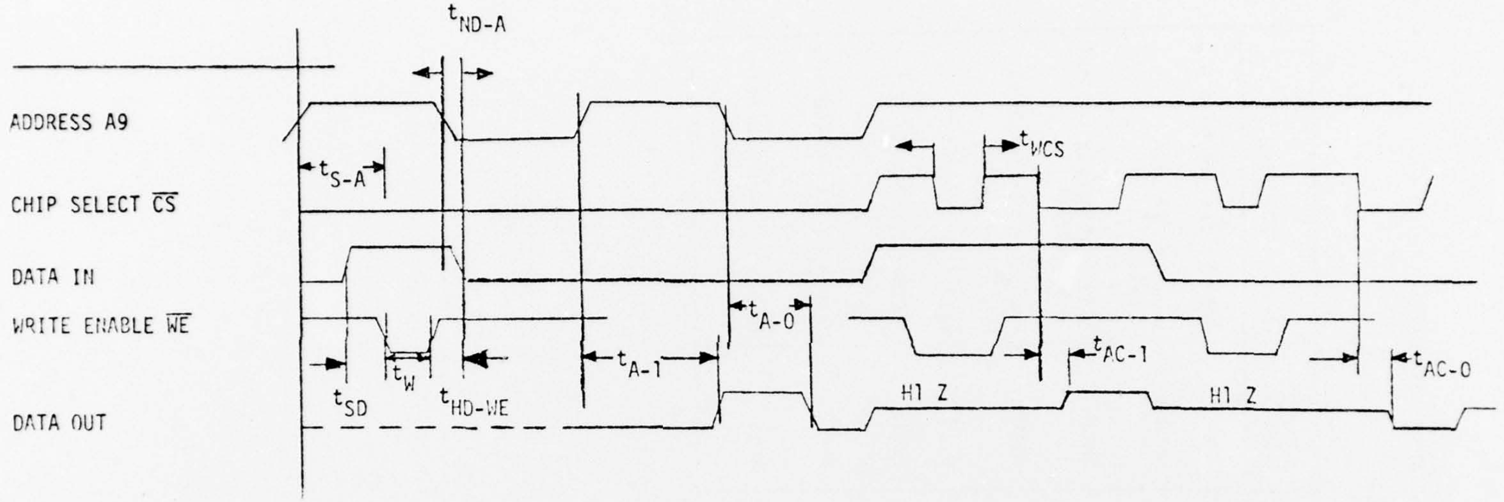
The 256 x 1 CMOS/SOS random access memory designated the TCS-041 was initially designated for implementation in the PWP. However, a stepped-up schedule on a commercial 1024 x 1 RAM was adopted by the RCA Solid State Division which met the time frame of the PWP. The use of a 1024 x 1 RAM rather than a 256 x 1 RAM reduces the number of RAM circuits in the PWP reorder memory from 220 to 176. Similarly, the number of modules required is reduced by 4. The foregoing factors, together with the attendant cost savings, led to the selection of the 1024 x 1 RAM for the PWP.

Problems with yield were experienced with the 1024 x 1 RAM at the time deliveries were due for the PWP. These problems have now been solved, but this, together with the circuit instability problem discussed in Section 4.2, led to the decision not to implement the reorder memory which uses the RAM's.

Since the intended application of the 1024 x 1 RAM occurred during the start-up phase of the commercial production, full device characterization was not available and operating parameters were developed.

Three circuits were tested, one of which developed an output drive problem prohibiting complete testing. An eight step operating sequence given in Table 24 was used to measure the minimum write cycle using the write enable control. The word locations used alternated between a 1 and 0 for address bit number 9 with all other address bits held constant. Address bit 9 presented a worst case condition for the address decode times. The results are given in Figure 35 together with the basic waveforms. The data indicate that a read/write cycle time of less than 100 nsec would be difficult at V_{DD} = 12 volts. In the write cycle mode with the chip select control held on, the minimum write cycle would be t_{S-A} (24 nsec) + t_W (35 nsec) + t_{HD-WE} (17 nsec) = 76 nsec plus timing register delays. The minimum read cycle is 80+ nsec.

An alternate mode of operation is to strobe the chip select control to write while the write enable control is on. This method shown in the second portion of the Figure 35 timing diagram permits simultaneous clocking of the data and address signals. This usually simplifies a memory timing and control system. An increase in operating speed can not be expected in this mode and the data of Figure 35 verifies this.



WORST CASE CONDITIONS: V_{DD} = 12 VOLTS

WRITE PULSE WIDTH: t_W = 35 NSEC

SET-UP TIMES: t_{S-A} = 24 NSEC, t_{SD} = 4 NSEC

DATA HOLD-WRITE ENABLE: t_{HD-WE} = 17 NSEC

DATA HOLD-WRITE/ADDRESS: t_{HD-A} = 3 NSEC

ACCESS TIMES: t_{A-0} = 55 NSEC, t_{A-1} = 80 NSEC, t_{AC-1} = 36 NSEC, t_{AC-0} = 30 NSEC

CHIP SELECT WRITE PULSE: t_{WCS} = 40 NSEC

FIGURE 35. TA-6780 1024 X 1 RAM MEASUREMENTS

TABLE 24
WRITE CYCLE TEST SEQUENCE

STEP	ADDRESSED WORD	READ/ WRITE	FUNCTION
1	A	Read	Read Content of Word A
2	B	Write "1"	Write a "1" Into Location B
3	A	Read	Read A to Verify Content is Unchanged
4	B	Read	Read B to Verify Write Operation (2)
5	A	Read	Read A
6	B	Write "0"	Write a "0" Into Location B
7	A	Read	Read A to Verify Content is Unchanged
8	B	Read	Read B to Verify Write Operation (6)

SECTION V

LSI PACKAGING DEVELOPMENT

The use of CMOS/SOS LSI circuit technology required the development of new packaging techniques. Ceramic hybrids provide the required low capacitance, high density interconnections for the CMOS/SOS circuits. However, a primary need was the development of a fabrication technique which would provide a high yield at the hybrid level and the large number of interconnections with the LSI circuits complicated this problem. In addition, CMOS/SOS technology had not been transferred to large scale hardware use; thus wiring rules tailored to the technology had to be developed.

5.1 CHIP CARRIER HYBRID PACKAGING

5.1.1 Selection of Chip Carriers

A basic problem in the mounting of large complex LSI devices on ceramic hybrid substrates is the low hybrid yield if traditional wire-bonding techniques are used. This effect is illustrated in Figure 36. The figure plots final hybrid package yield versus the number of devices with individual chip yield as the third variable. If individual chips are mounted by the wire-bonding technique, a chip yield of 90% would be good under the conditions of an advanced technology LSI component. The number of devices on the PWP module will be from seven to sixteen giving a total module yield of only from 10% to 50% for the 90% chip yield case. However, if the chip yield is around 98%, as would be expected with packaged, tested products, the module yield increases to about 90% or better. This hybrid yield effect was the over-riding factor in the decision to package each individual LSI circuit in Alsipak ⊛ chip carriers. The chip carriers offer a compromise between the efficient but developmental beam lead fabrication-assembly technique and standard large commercial packaging. The LSI chips are mounted in the carriers by conventional bonding techniques. The carriers are then hermetically sealed and tested. The tested circuits are attached to the ceramic hybrid substrate by re-flow soldering.

In addition to the initial hybrid yield improvement which reduces fabrication cost while increasing reliability, the chip carrier approach improves maintainability. A defective device is replaced much more easily than if conventional device attachment procedures are used.

5.1.2 Module Description

An important consideration for the PWP was the selection of the module size and the number pins. Examination of the various partitioning alternatives for the PWP indicated that the most efficient package from the point of view of the system architecture should have over 88 signal pins to encompass the full radix-2 I/O pipeline. However, this would impose a requirement for at least a total of 110 I/O pins and no connectors of this capacity were available. A partitioning approach was derived which permitted the system to be efficiently modularized with a total module pin count of 80.

⊛ Registered Trademark of 3M Company

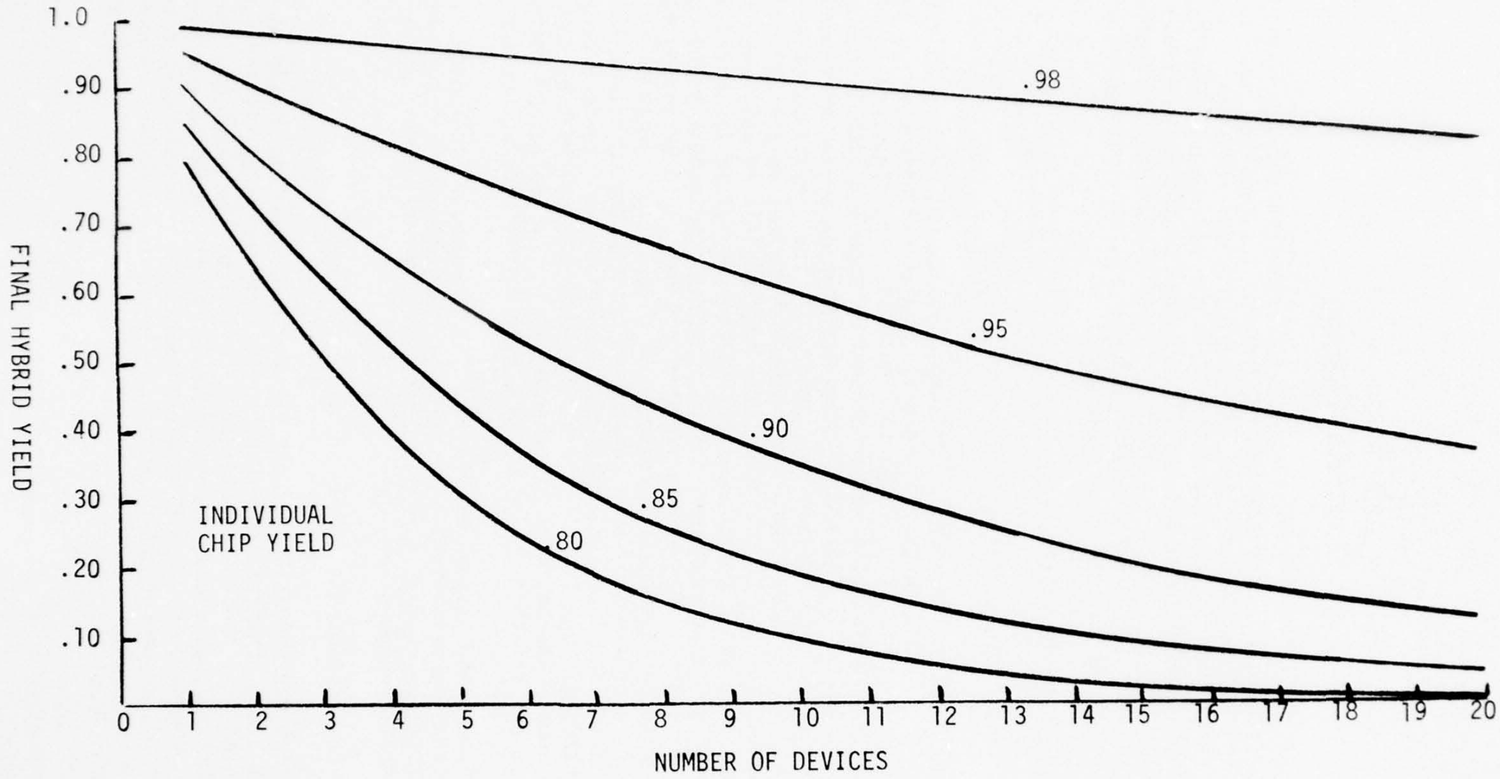


FIGURE 36. HYBRID YIELD VERSUS THE NUMBER OF COMPLEX LSI CHIPS PER HYBRID FOR VARIOUS INDIVIDUAL CHIP YIELDS

Internal development work at RCA had provided a ceramic hybrid packaging concept called HYPAK. The basic HYPAK design is shown in Figure 37, and contains a 1.0 in. x 2.0 in. ceramic circuit substrate with a forty pin connector. The HYPAK modules mount on 0.2 in. pitch.

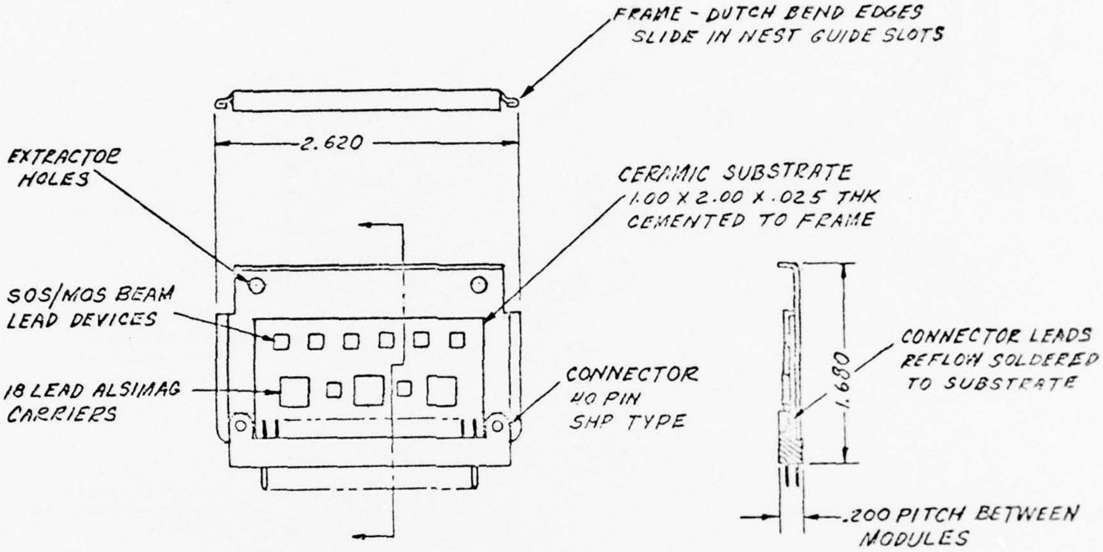


FIGURE 37. HYPAK MODULE WITH BEAM LEAD DEVICES

The HYPAK module was designed to mount all beam lead devices on ceramic substrate. However, they will accommodate ceramic chip carriers. These carriers are up to 0.5 in. square and consume too great an area to allow efficient partitioning of all circuits on the HYPAK module. In addition, the number of pin interconnections required exceed those available on the HYPAK module. If CMOS/SOS beam leaded devices and a high capacity connector were available, there would then be no area or pin problem in packaging the elements of the PWP on the HYPAK module.

A double length HYPAK module form was selected for the PWP since it permitted sufficient pin capacity and space for mounting the chip carriers. The standard double length HYPAK module consists of a 1.4" x 5" alumina substrate mounted on a metal frame as shown in Figure 38. There are two 40 pin connectors (J1 and J2) on either side of the bottom of the module. The connector makes contact with the substrate by means of fingers on 50 mil pitch.

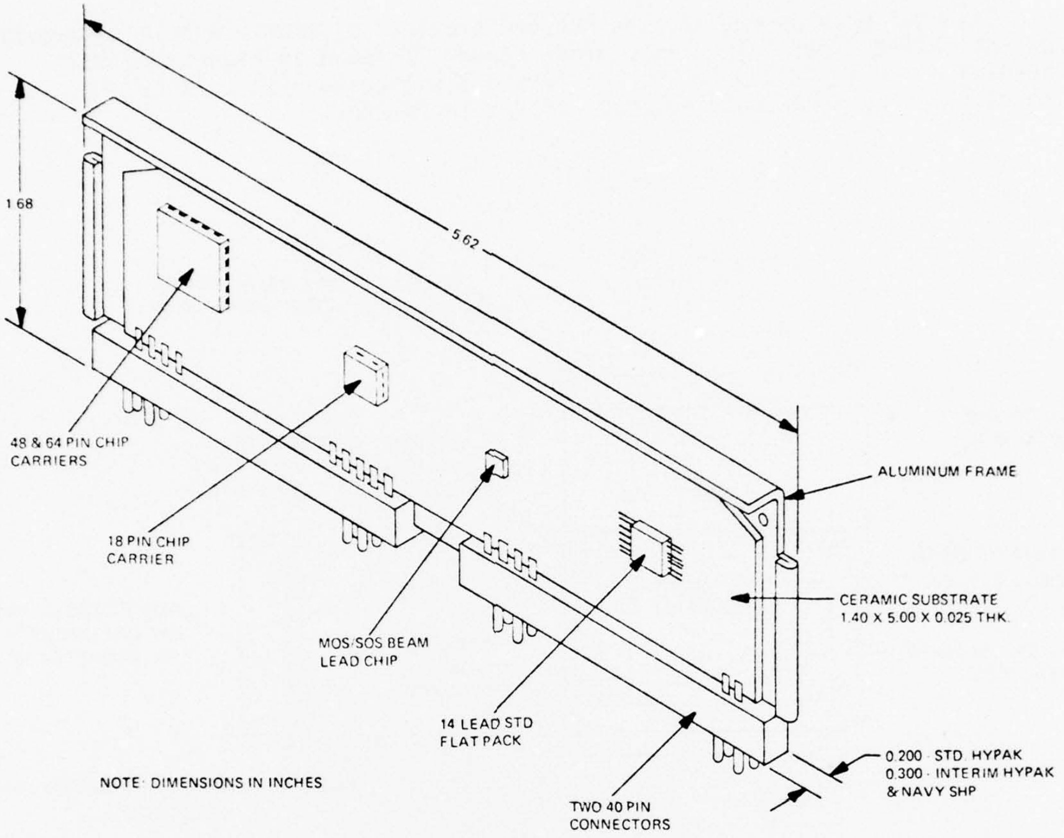

1.68
5.62
48 & 64 PIN CHIP CARRIERS
18 PIN CHIP CARRIER
MOS/SOS BEAM LEAD CHIP
14 LEAD STD FLAT PACK
ALUMINUM FRAME
CERAMIC SUBSTRATE 1.40 X 5.00 X 0.025 THK.
0.200 - STD HYPAK
0.300 - INTERIM HYPAK & NAVY SHP
TWO 40 PIN CONNECTORS
NOTE: DIMENSIONS IN INCHES

FIGURE 38. HYPAK MODULE - DOUBLE LENGTH

On the substrate, there are two sets of 40, 30 X 100 mil pads on 50 mil center-to-center spacing to accommodate these fingers.

Power and ground pins on the connector are dedicated to make the module as compatible with the Navy's Standard Hardware Program (SHP) as possible. This is outlined in the chart below.

		DOUBLE HYPAK	
	SHP	J1	J2
+5V	1 (21)		1
+15V		1	
-5V	20 (40)		
GROUND	10 (30)	2,6,10,15,19	2,6,10,15,19
FRAME GROUND	11 (31)		

Numbers in parenthesis are OPTIONAL

In the double HYPAK module, the principle power supply voltage is brought through pin 1 of J1. If there is mainly a TTL System, +5V would be brought through this pin. In the case of a mixed logic system, +5V is brought through pin 1 of J2 which is compatible with a single length SHP module. Pins 20 and 40 of both J1 and J2 are to be left open for any auxiliary power or system clocks, if necessary. Preferably, pins 20 and 40 of J2 handle phase 1 and phase 2 clocks respectively. The double HYPAK module has a total of 10 ground pins evenly spaced over the connector to lower the inductance of the signal return path.

5.1.3 Nests and Backplane

The nests for both the single and double HYPAK are similar in construction. Formed sheet metal slots act as guides for the modules and also provide air passages through the closely stacked modules.

The nests for both type modules mount on a common wire wrap backplane surface allowing automatic machine wirewrapping of all interconnecting wiring. Ribbon cable connectors for interfacing with A/D's, central computers and other equipment, plug into the backplane in the same manner as do the modules. The nest frames have mounting flanges for mounting onto standard 19.00 in. EIA frames.

5.2 CMOS/SOS DESIGN RULES

5.2.1 Overall Consideration

Rules based on laboratory measurements were developed which enable calculation of capacitive loading on output gates and potential crosstalk problems. The rules are presented in as general a way as possible so that they can be adapted to various cases of dielectric thickness, line spacings and device capabilities. The power distribution rules apply specifically for the standard double length HYPAK module used in the PWP, but can be adapted to other module types.

5.2.2 Module Power Distribution and Decoupling

The rules for the layout of the power bus and decoupling on the thick film modules are based on the following assumptions:

1. For CMOS/SOS devices, the device current is approximated by a uniform current pulse of 20 nanoseconds duration. The standby current is assumed to be negligible. The total device dissipation, therefore, is assumed to be the resultant average of the series of uniform pulses at the operating repetition rate.

2. The module power bus, decoupling capacitors and IC tap on the bus were sized to provide approximately equal drops for all CMOS/SOS modules independent of power dissipated. The average drop is budgeted at about 0.5 volts with a peak drop of 0.75 volts.

3. The main power distribution bus to the modules in a nest is assumed to have the following characteristics:

PARAMETER	BULK CHARACTERISTICS
R	0.00022 Ohms/foot
L	4.4 nanohenries/foot
C	490 picofarads/foot
Z_0	3 Ohms (Approximately)

4. The resistivity of the conductive inks used in the thick film process is 0.02 ohms per square for modules using CMOS devices.

5. The bus layout for TTL devices is similar to that for CMOS/SOS devices except the drop is budgeted at 0.05 volts.

6. The resistivity of the conductive inks used in the thick film process is 0.001 ohms per square for modules using TTL devices.

5.2.3 Design Rules

The power bus for CMOS modules shall be routed around the perimeter of the module as shown in Figure 39.

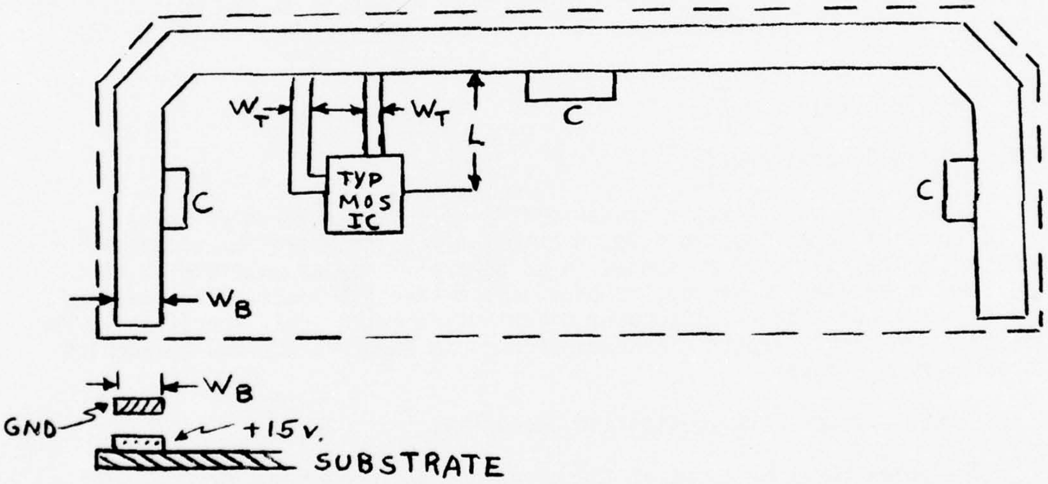


FIGURE 39. POWER BUS ON HYPAK MODULE

The width of the bus (W_B) is determined from the following expression where P_d is total module power in watts:

$$W_B = P_d \, (22.88) \text{ MILS}$$

The width of the IC tap (W_T) is a function of the location of the IC (L) from the main bus and is determined from the following expression where P_d is the IC dissipation in watts:

$$W_T = \frac{P_d \, L}{18.75}$$

Three capacitors (C) are located along the bus as shown in Figure 39. The capacitors are 0.1 microfarads, ± 10%, 50 V such as Varadyne #5BX050S104K.

5.2.4 Special Considerations

High current devices such as clock drivers require additional decoupling if they are not located near one of the existing capacitors on the board. Each case is evaluated in terms of peak load current and allowable supply voltage drop to determine the line widths and decoupling required to maintain supply tolerances.

Additional supply voltages may also be required by the devices which must be decoupled. Depending on individual circuit characteristics and loading, a minimum of one decoupling capacitor must be included for each additional supply.

5.2.5 Capacitance Measurements

In order to obtain data from which wiring rules can be generated, a series of test substrates were built.

The capacitance between two parallel lines in Table 25 was measured for various line widths and spacings and various thicknesses of dielectric covering the lines. The 5 mil lines are set on a 10 mil grid, the 8 mil lines are on a 15 mil grid and the 10 mil lines are on a 20 mil grid. The spacings are multiples of these standard grids. For an example, a 5 mil line on 10 mil center-to-center spacing is designated as a 5 mil line on 1X spacing; an 8 mil line on 30 mil center-to-center spacing is designated as an 8 mil line on 2X spacing.

TABLE 25
CAPACITANCE BETWEEN PARALLEL LINES

CAPACITANCE BETWEEN TWO PARALLEL LINES

		DIELECTRIC OVER LINES - MILS				
WIDTH MILS	SPACING	0	1.4	2.4*	3.4	6.6*
5	1X	1.9	2.0	2.3	2.6	2.9
5	2X	1.3	1.4	1.8	1.9	2.2
8	1X	1.9	2.0	2.6	2.8	3.1
8	2X	1.3	1.4	1.7	2.1	Δ
8	3X	1.1	1.2	1.5	1.9	1.9
8	4X	0.9	0.9	1.3	1.7	Δ
10	1X	1.8	1.9	2.3	3.0	3.3
10	2X	1.2	1.2	2.0	2.4	Δ

* - ESTIMATE
Δ - INSUFFICIENT DATA

Adding a thin dielectric coating does not increase capacitance appreciably until its thickness exceeds 1.5 mils.

The capacitance of one line to its neighbors is given in Table 26. All of the cases are of one line to its immediate neighbor on either side of it except for that marked "+" which is the capacitance to the two lines on either side of it. Again, all of the capacitances are in pf/in.

Note that in the case of 8 mil lines at 1X spacing the capacitance between a line and its immediate neighbors and that line and its two immediate neighbors is almost identical. This indicates that the capacitance between one line and all other lines is essentially that between it and its immediate neighbors.

The capacitance of a line to a ground plane appears to be relatively independent of the dielectric thickness over the line. For a 5 mil line, the capacitance is 1.8 pF/in, for an 8 mil line, 2.0 pF/in, and a 10 mil line, 2.3 pF/in.

TABLE 26
CAPACITANCE TO NEIGHBORING LINES

WIDTH MILS	SPACING	DIELECTRIC OVER LINES - MILS 0	1.4	2.4*	3.4	6.6*
5	1X	2.9	3.1	2.5	3.7	4.4
8	1X	3.3	3.5	3.8	4.0	4.9
8	1X +	3.4	3.7	Δ	4.0	Δ
8	2X	2.0	2.3	Δ	2.6	3.0
10	1X	3.1	3.4	Δ	4.0	Δ

* - ESTIMATE
Δ - INSUFFICIENT DATA

The capacitance due to the crossing of lines is due mainly to the fringing fields. A straightforward application of the parallel plate formula yields crossover capacitances that are much lower than the actual measured values. Measurements were made of crossovers in three cases; lines on a close spacing crossing lines on a close spacing, lines on a close spacing crossing lines on a far spacing, and lines on a far spacing crossing lines on a far spacing. The term "close spacing" means that the lines are set on their standard grid as defined previously. "Far spacing" indicates that the lines are on 100 mil center-to-center spacing. Table 27 gives a compilation of the data generated for the three cases.

TABLE 27
CROSSOVER CAPACITANCE

TYPE OF CROSSOVER MIL X MIL	CLOSE SPACING/ CLOSE SPACING pF/CROSS.	CLOSE SPACING/ FAR SPACING pF/CROSS.	FAR SPACING/ FAR SPACING pF/CROSS.
5X5	.13	Δ	.33*
8X8	.14	.13	.28*
10X10	.18	.19	.39*
8X20	Δ	Δ	.48*
10X20	Δ	Δ	.43

* - ESTIMATE
Δ - NO DATA AVAILABLE

The thickness of the dielectric between the lines for all cases is 3.4 mils with $\epsilon_r \approx 11$. Notice that the capacitance per crossover is about the same for the close/close spacing and close/far spacing, but is much higher for the far/far spacing. This indicates that the capacitance is indeed due largely to fringe fields.

5.2.6 Crosstalk

The propagation delay down an 8 mil line is ≈ 220 ps/in as measured using a time domain reflectometer. Since the rise times involved in the CMOS/SOS technology is 5 ns or greater, a lumped parameter model for crosstalk can be used. The equivalent circuit is given in Figure 40. R_o represents the output impedance of a gate driving the passive line and C_L represents the total capacitive load on the passive line due to gate inputs. C_S is the total amount of coupling capacitance between an active line and the passive line. Measurements have been made on crosstalk using the critical test substrate which verify the above model. If the output waveform of a gate is an exponential, $v_o(t) = V\,[1\text{-EXP}(-t/\tau)]$, the percent crosstalk is given by:

$$\%\ \text{CROSSTALK} = \frac{100\, R_o\, C_S}{\tau - R_o(C_S+C_L)} \left\{ \left[\frac{R_o(C_S+C_L)}{\tau}\right]^{-\frac{R_o(C_S+C_L)}{R_o(C_S+C_L)-\tau}} - \left[\frac{R_o(C_S+C_L)}{\tau}\right]^{-\frac{\tau}{R_o(C_S+C_L)-\tau}} \right\}$$

where τ is defined as the time constant of the exponential.

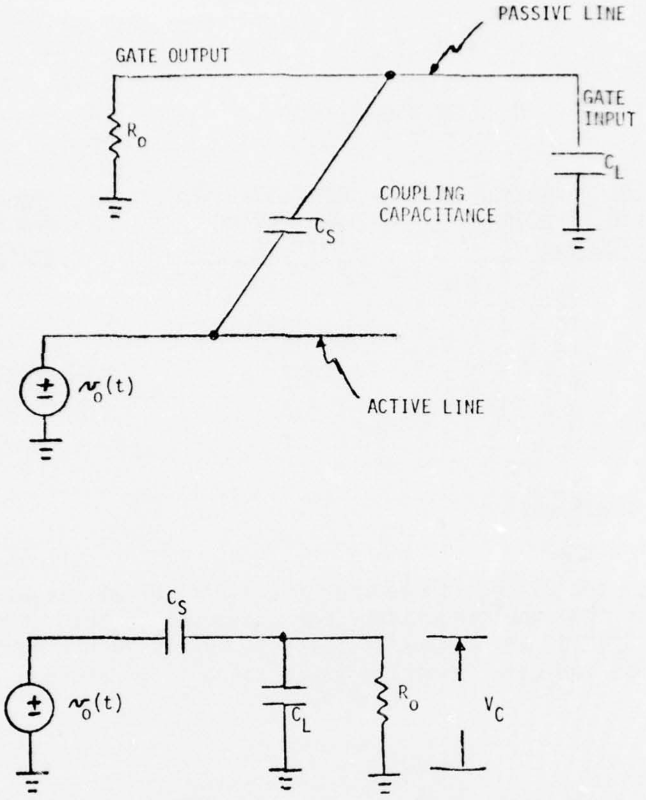


FIGURE 40. CROSSTALK MODEL

If the lines are driven by a source whose output is characterized as a ramp,

$$v_o(t) = V \left[\frac{t}{a} u(t) - \frac{(t-a)}{a} u(t-a) \right] ,$$

as with a clock driver, the percent crosstalk is given by:

$$\text{\% CROSSTALK} = \frac{100 \ C_S R_o}{a} \quad 1 - \exp \left[- \frac{a}{R_o (C_S + C_L)} \right]$$

On a module, lines will be driven from sources with different output impedances and different rise time capabilities. In the event of a significant amount of coupling between lines, each case must be analyzed separately. Depending on the rise time of the signal on the active line and the output impedance of the gate on the passive line, varying amounts of coupling capacitance will be allowed. For CMOS/SOS circuits, the crosstalk is limited to 30%. A set of curves is plotted on Figures 41 and 42. They show the maximum amount of coupling allowed for various cases of rise time and output impedance. A fanout of one (C_L = 2 pF) is assumed on the passive line. This represents a worst case. A larger fanout will lower the crosstalk, but also raise the total capacitance to be driven by that particular gate which will lower the line length due to a restriction of total capacitance on the gate. Any combination of parameters falling below a particular curve represents an allowed condition.

Crosstalk between lines over a ground plane should present no problem. A case with more than four inches of parallelism with the possibility of two lines straddling a passive line switching simultaneously should be avoided. In such a case, the lines can be simply spread further apart or located elsewhere. This is the worst case condition.

5.2.7 Design Rules for Signal Lines

A general method for determining if a particular signal path is allowable, is outlined below:

1. Determine the drive capability of the source. The amount of capacitance that the gate will drive with a given rise time is the major constraint put on a CMOS/SOS device.

2. From the maximum capacitance to be driven, subtract 2pF for each load to be driven in an "ALSIPAK" carrier and 5pF for each load to be driven in a flat pack. This gives the total amount of capacitance that can be budgeted for wiring.

3. The wiring capacitance due to neighboring lines and crossovers can be calculated from the data. The appropriate crossover value for the particular case, i.e., a closely spaced group of lines or far spaced lines, should be used.

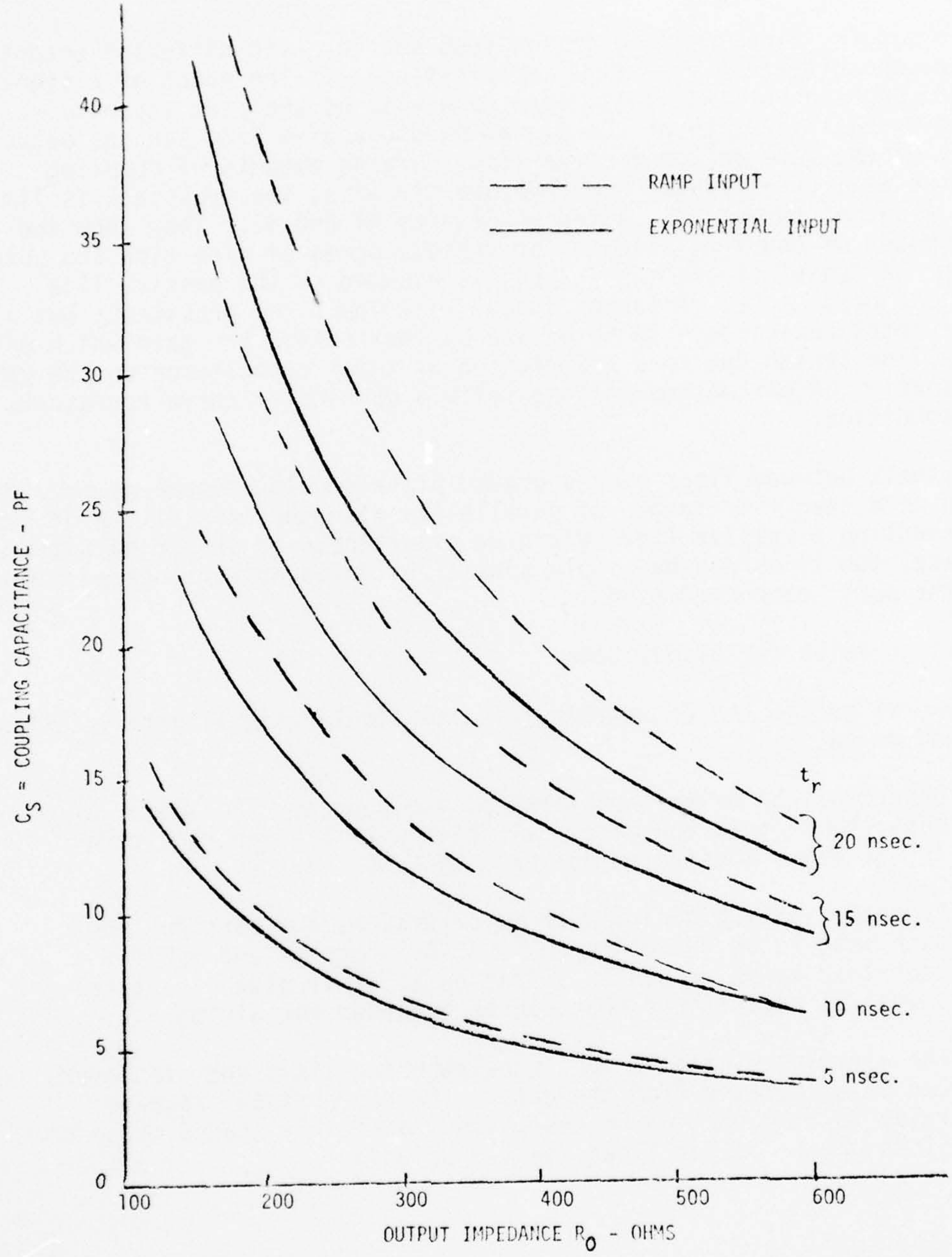


FIGURE 41. 30 PERCENT CROSSTALK LIMITS WITH C_L = 2 PF., COUPLING CAPACITANCE VERSUS OUTPUT IMPEDANCE

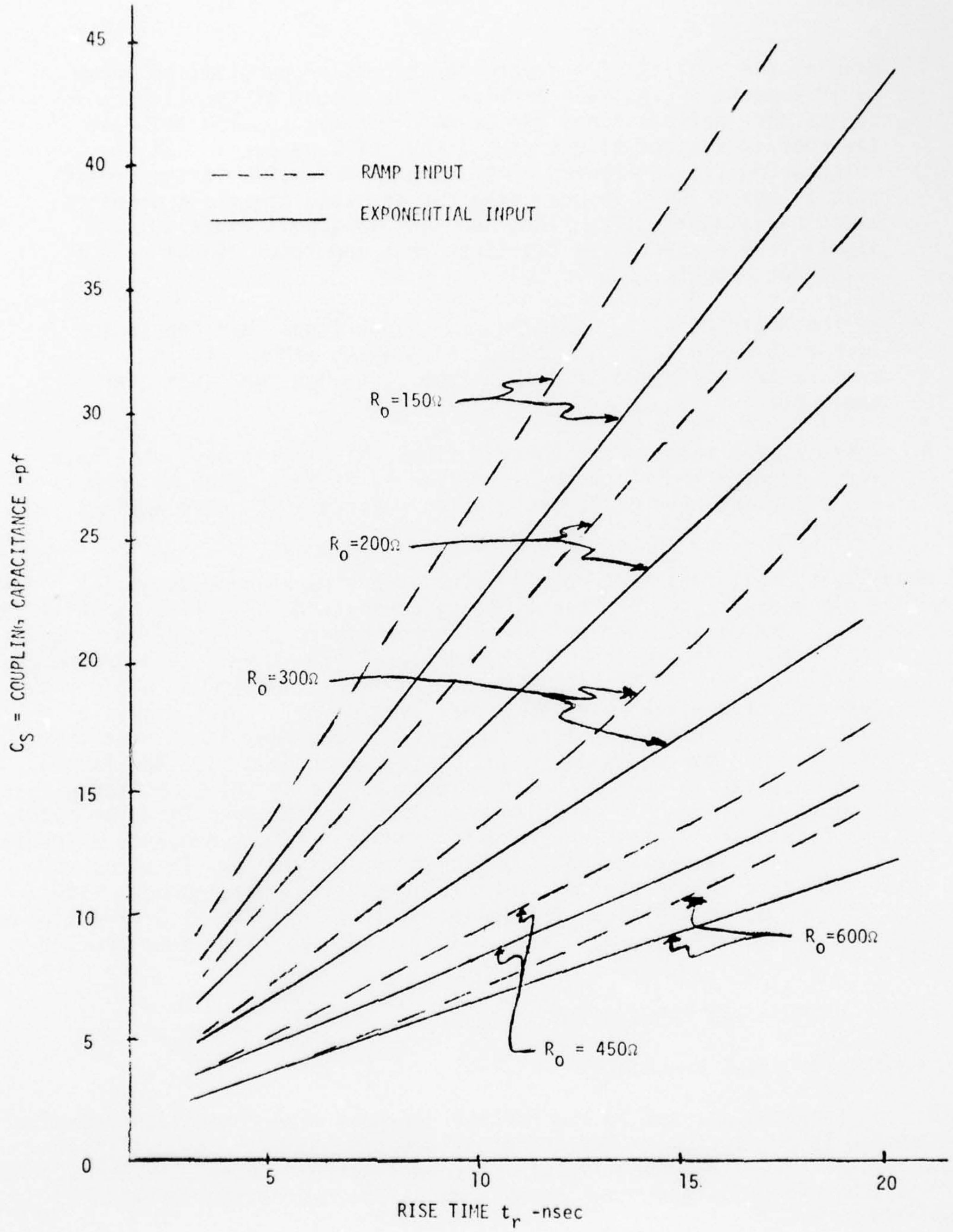
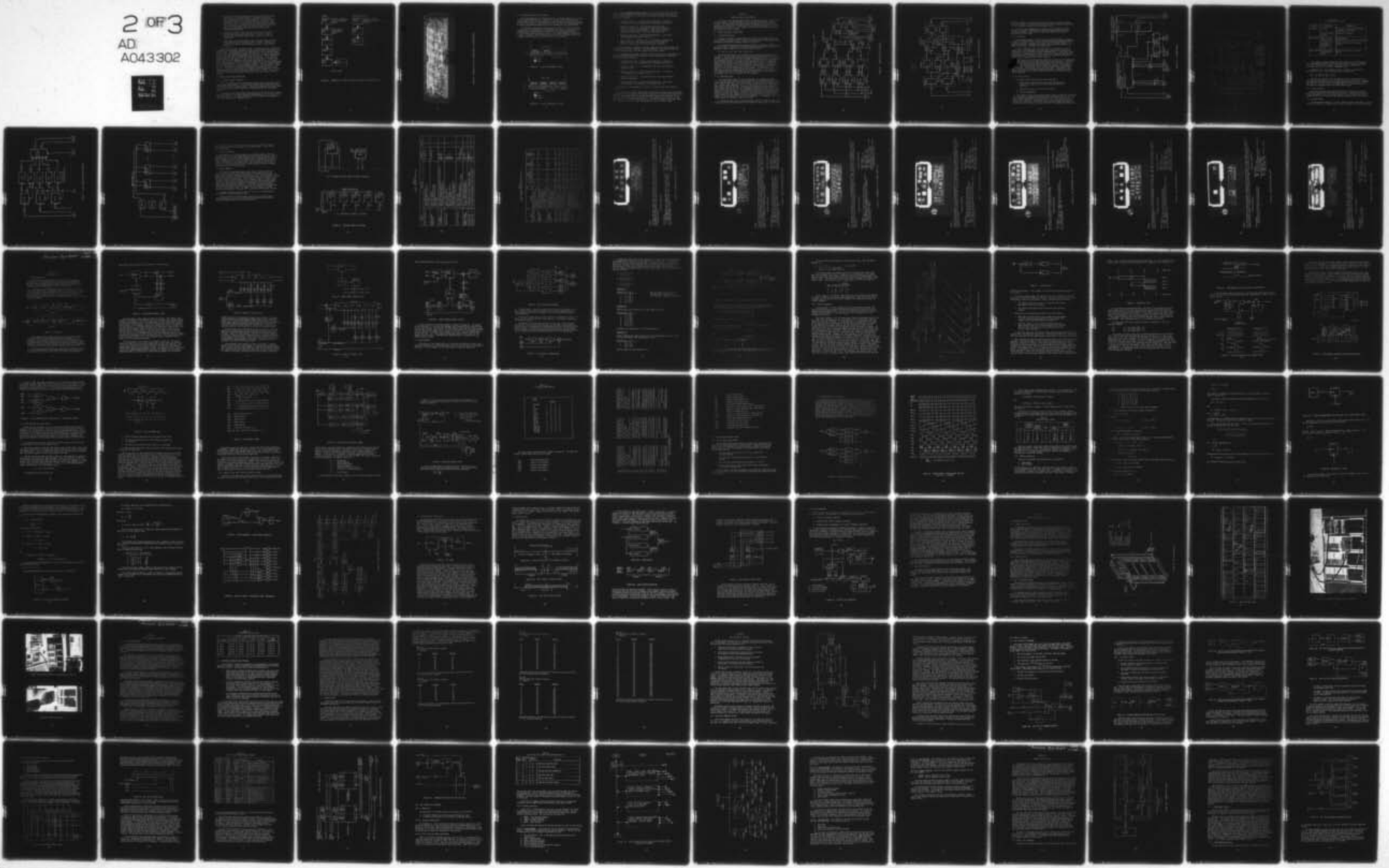


FIGURE 42. 30 PERCENT CROSSTALK LIMITS WITH C_L = 2 PF., COUPLING CAPACITANCE VERSUS RISE TIME

AD-A043 302 RCA GOVERNMENT SYSTEMS DIV MOORESTOWN N J MISSILE AND--ETC F/G 17/9
PROGRAMMABLE FFT LINEAR FM WAVEFORM PROCESSOR. PHASES II AND II--ETC(U)
JUL 77 L W MARTINSON, J A LUNSFORD F33615-74-C-1077

UNCLASSIFIED AFAL-TR-77-23 NL

2 OF 3
AD
A043302



4. Examine the routing of the path for excessive parallelism which could present a crosstalk problem. The amount of coupling capacitance between lines can be determined. Knowing this and the characteristics of the signal that will be present on the neighboring lines, Figures 41 or 42 will determine if the crosstalk is below 30%. In searching for possible crosstalk problems, avoid cases where clock lines run next to signal lines or a signal line straddled by two lines that are likely to be switching simultaneously.

5. Wiring Restrictions: Avoid running clock lines near inputs to latches because crosstalk could cause a bit error. Avoid running input lines over output lines. Do not run lines over one another.

6. There may be cases where control lines, or clock lines, will have large fanouts and an on module driver is needed. Examine such cases for the need of series damping resistors to guard against ringing.

When laying out lines that go off module, consideration must be taken for the module pin capacitance, backplane wiring capacitance and drive capabilities of the source. The capacitance of the connector pin in the backplane ranges from 2.3 to 2.5 pF depending on whether the signal pin is next to a ground pin or not. The capacitance of backplane wiring is from 1 to 2 pF/in depending on whether there are many wires near the signal line or not. In the case of a critical path, or the need for a long run on the backplane, it is best that other lines with low impedances to ground be kept away from it. The fanout onto a module must be limited to one in most cases due to the capacitance picked up by the connector pins. For an example, suppose a line runs from the middle of J1 to the middle of J2 on the neighboring module. This represents a length of 3.5". This could be considered a long path, so assume that it picks up (3.5 in) (1 pF/in) = 3.5 pF due to the backplane wiring capacitance. With a fanout of one on the module (2 pF), this leaves 15 pF - 3.5 pF - 2 pF - 5 pF = 4.5 pF for on module wiring capacitance which allows very little on module wiring.

5.3 MODULE LAYOUTS AND FABRICATION

5.3.1 Applicon Layout Procedures

All of the modules used in the PWP are designed with an Applicon computer-aided design-layout system. This technique greatly improves the quality and reduces the cost over a manual layout. The Applicon procedure eliminates three of six of the process steps required by a manual procedure as indicated by Figure 43.

An example of the artwork-design generated by use of the Applicon system is shown in Figure 44. The figure shows the interconnections of all layers of the FFT memory module. The actual plot from which the figure was reproduced has each layer presented as a separate color.

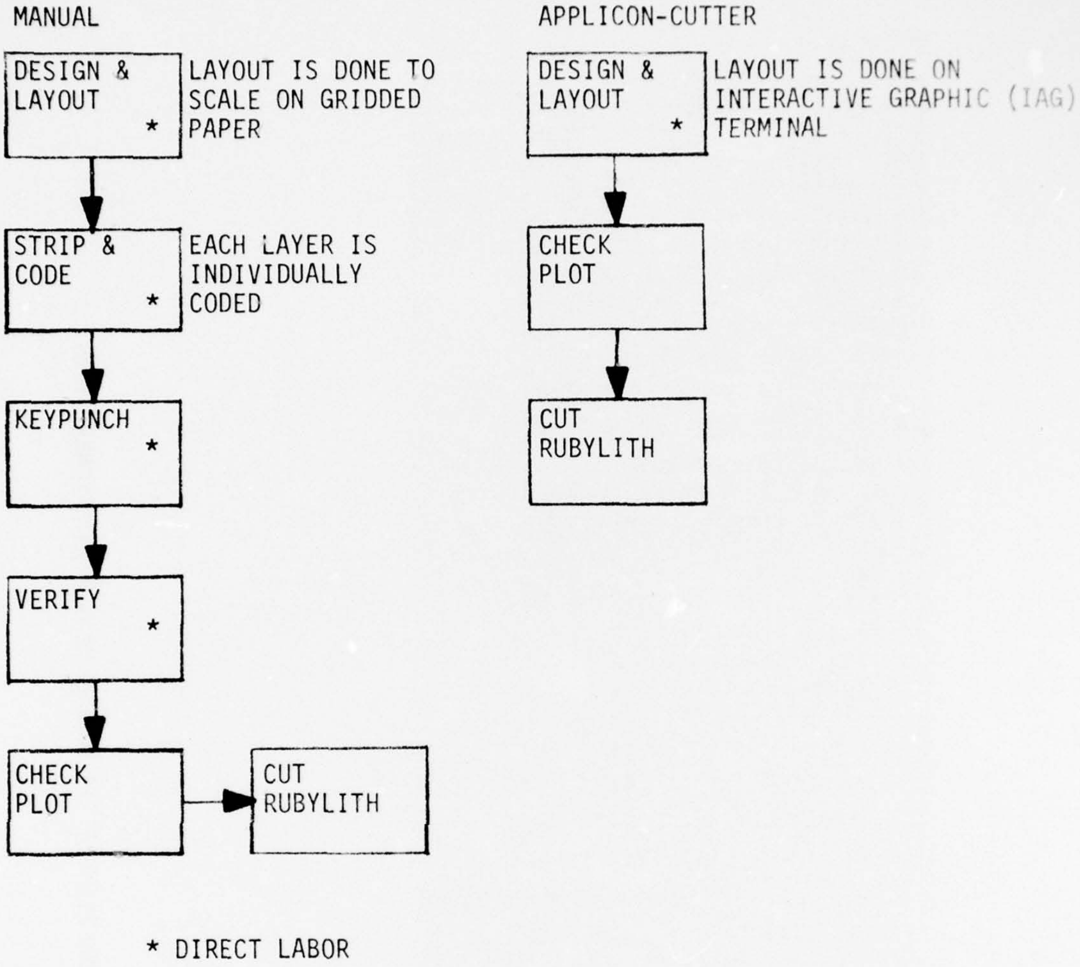


FIGURE 43. COMPARISON OF MANUAL AND APPLICON HYBRID DESIGN PROCEDURES

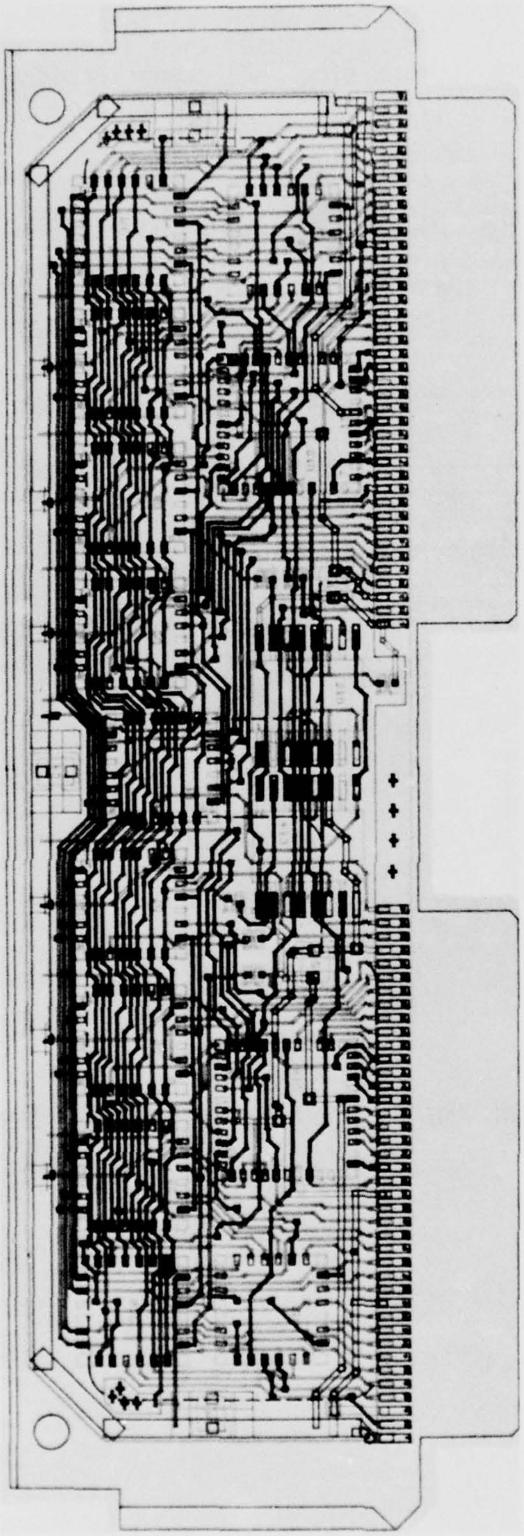


FIGURE 44. COMBINED PLOT OF FFT MEMORY MODULE INTERCONNECTIONS

5.3.2 Module Fabrication and Assembly

The primary approach for attaching devices to the PWP module will be reflow soldering of ALSIPAK Carriers, conventional surface attach flatpacks and a minimum number of ceramic chip capacitors. The projected size of the substrate is 5" x 1.4" x .025". A maximum of 13 ALSIPAK carriers are incorporated on the FFT memory module design and a maximum of 8 flatpacks on the control module.

Two different interconnect techniques are utilized. The first consists of two layers of interconnect with the ALSIPAK carriers and flatpacks reflow soldered to the second layer as in Figure 45. This method was utilized to interconnect moderate density circuits. The second method for high density circuits utilized three interconnect layers with the ALSIPAK carriers and flatpacks reflow soldered to the third layer as in Figure 46.

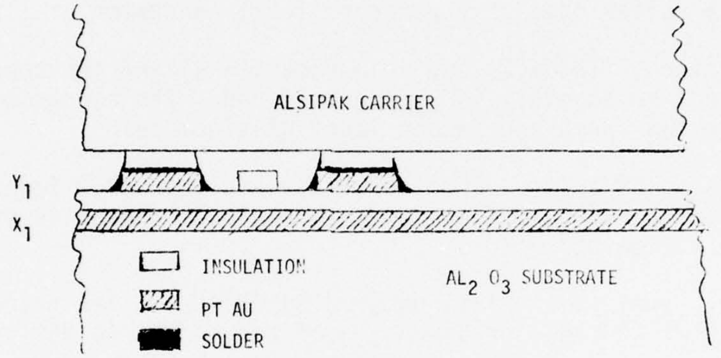


FIGURE 45. 2 LEVEL INTERCONNECT ALSIPAK

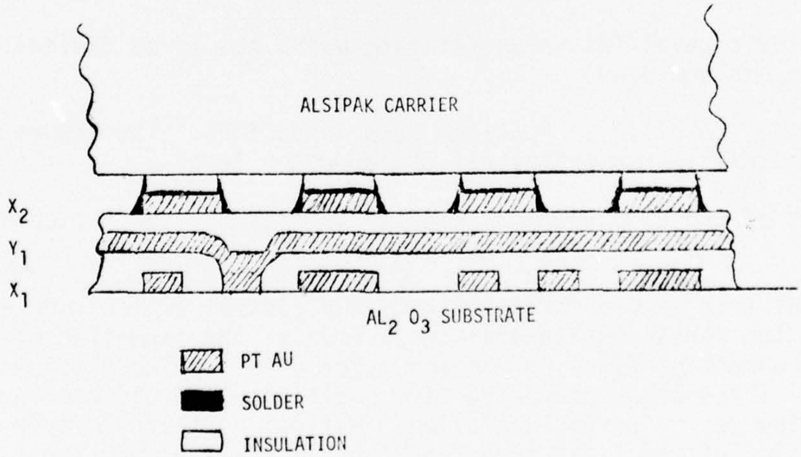


FIGURE 46. 3 LEVEL INTERCONNECT ALSIPAK

5.3.2.1 Low to Moderate Density Circuits - Initial examination of the interconnect drawing determines if the circuit can be fabricated utilizing two layers of interconnects. For two layer interconnect systems, the following levels are assigned.

1. Conductor Level (1). Platinum Gold Conductors. The traces run parallel to the long dimension of the module (4.950 in).

2. Dielectric Level (1). Two separate print and fired dielectric layers are utilized. If the circuit density is light, a multilayer construction is modified to minimize intralayer capacitance, to utilize cross-overs only.

3. Conductor Level (2). Platinum Gold Conductors. The traces run orthogonal to conductor level (1).

4. Solder Level (1). Solder Level. DIP Soldering conventional 60/40 solder or solder paste screening will be used to deposit a solder coat on the second level conductor.

Devices (carriers, flatpacks and chip capacitors) and the connector are fixtured and the entire assembly is reflow soldered. The connector pads are screened with both the first and second level platinum gold.

5.3.2.2 High Density Circuits - When more than two levels of interconnect are required because of the circuit density, a three level system is employed consisting of the following:

1. Conductor Level (1). Platinum Gold Conductors. The traces run parallel to the long dimension of the module (4.950 in.).

2. Dielectric Level (1). Two separate print and fired dielectric layers are used.

3. Conductor Level (2). Platinum Gold Conductors. The traces run orthogonal to conductor level 1.

4. Dielectric Level (2). Two separate print and fired dielectric layers are employed.

5. Conductor Level (3). Platinum Gold Conductors. The traces will generally run orthogonal to conductor level 2.

Attachment of the components is similar to that for the two conductor system.

The initial test sample fabrication of the control switch module uncovered a problem with the foregoing fabrication technique; the formation of small solder bumps on interconnection paths on the top layer prior to the placement of the chip carriers. These bumps prevented firm positioning of the carriers on the substrate for the reflow solder operation. A final dielectric layer was, therefore, added on top of the final interconnection for all module types.

SECTION VI

FUNCTIONAL MODULE DEVELOPMENTS

A total of eight special module designs are used in the PWP, including two universal modules, one for CMOS/SOS circuits and the other for TTL control circuits. This low number has been achieved by careful partitioning of the system and by making many of the modules serve multiple functions. All of the modules have the same 1.7" x 5.6" physical size with 80 input/output pins and are mounted on 0.3 inch spacing.

6.1 MODULE FUNCTIONAL DESCRIPTIONS

6.1.1 Complex Multiplier

The most functionally complex module in the PWP is the complex multiplier which serves as a vector rotator in the FFT's and provides the de-ramping, phase correction and weighting functions external to the FFT's.

The functional diagram of the complex multiplier is shown in Figure 47. It consists of four TCS-057 9x9 bit sign-magnitude multipliers which form the four magnitude products required for the complex multiplication.

$$(a + jb)\ (c + jd) = (ac - bd) + j(ad + bc)$$

The multiplier outputs are rounded to 8 bits plus sign and fed to retimer registers. The multiplier outputs are converted to signed products in the retimers by inputting the sign to the retimer complement control. The bd product must be subtracted and its sign is, therefore, inverted prior to being tied to the complement control. The output of the retimers are 9 bit 1's complement numbers which are inputted to the TCS-065 adder arrays where 1's complement addition is performed. If an overflow occurs in either adder, the output of both adders is left shifted one bit, the last bit is truncated and an overflow signal is sent to the floating point logic control. In the final retimer provision is made for conjugating the output by bringing out the inverted sign and complement control of the imaginary (Q channel) output. Similar outputs are brought out of the real channel for test purposes.

6.1.2 Adder/Subtractor

The second module making up the FFT arithmetic is an adder/subtractor unit. The required functions to be performed are the addition and subtraction of two complex data words. The total number of signal inputs and outputs to a full, complex adder-subtractor is 4 x 22 = 88 for the 22 bit (9I, 9Q 4 exponent) complex words used in the PWP. Packaging the add/subtract module either as a full complex word (88 I/O pins) or as separate complex adders and subtraction (66 I/O pins plus controls) is not realizable on the 80 pin module with 12 pins dedicated to power and ground. The partitioning selected for the add/subtract module is to split the inputs into their I and Q components and obtain the sum and difference of each component on separate modules. In this way, the basic number of I/O pins required is only 9 x 4 (mantissas) + 16 exponents = 52. This allows 18 pins for controls of which 9 are used.

A functional diagram of the adder/subtractor module is shown in Figure 48. Based upon the relative magnitudes of the respective inputs, the I_R (from complex

FUNCTION: $(a+jb)\ (c+jd) = (ac-bd) + j(ad+bc)$
$= I_R + jQ_R$

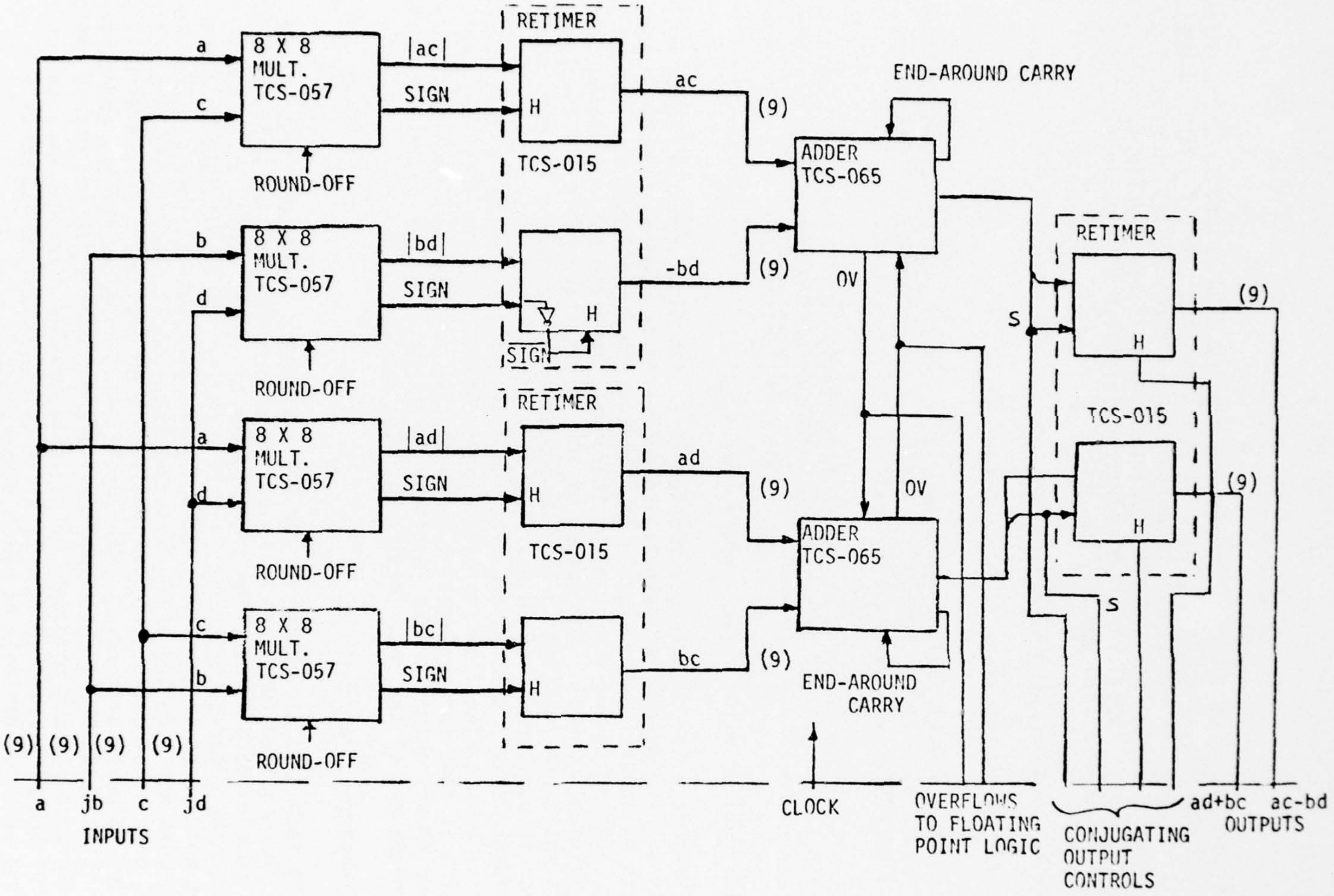


FIGURE 47. COMPLEX MULTIPLIER MODULE

$$\Sigma_I = I_A + I_R$$

$$\Delta_I = I_A - I_R$$

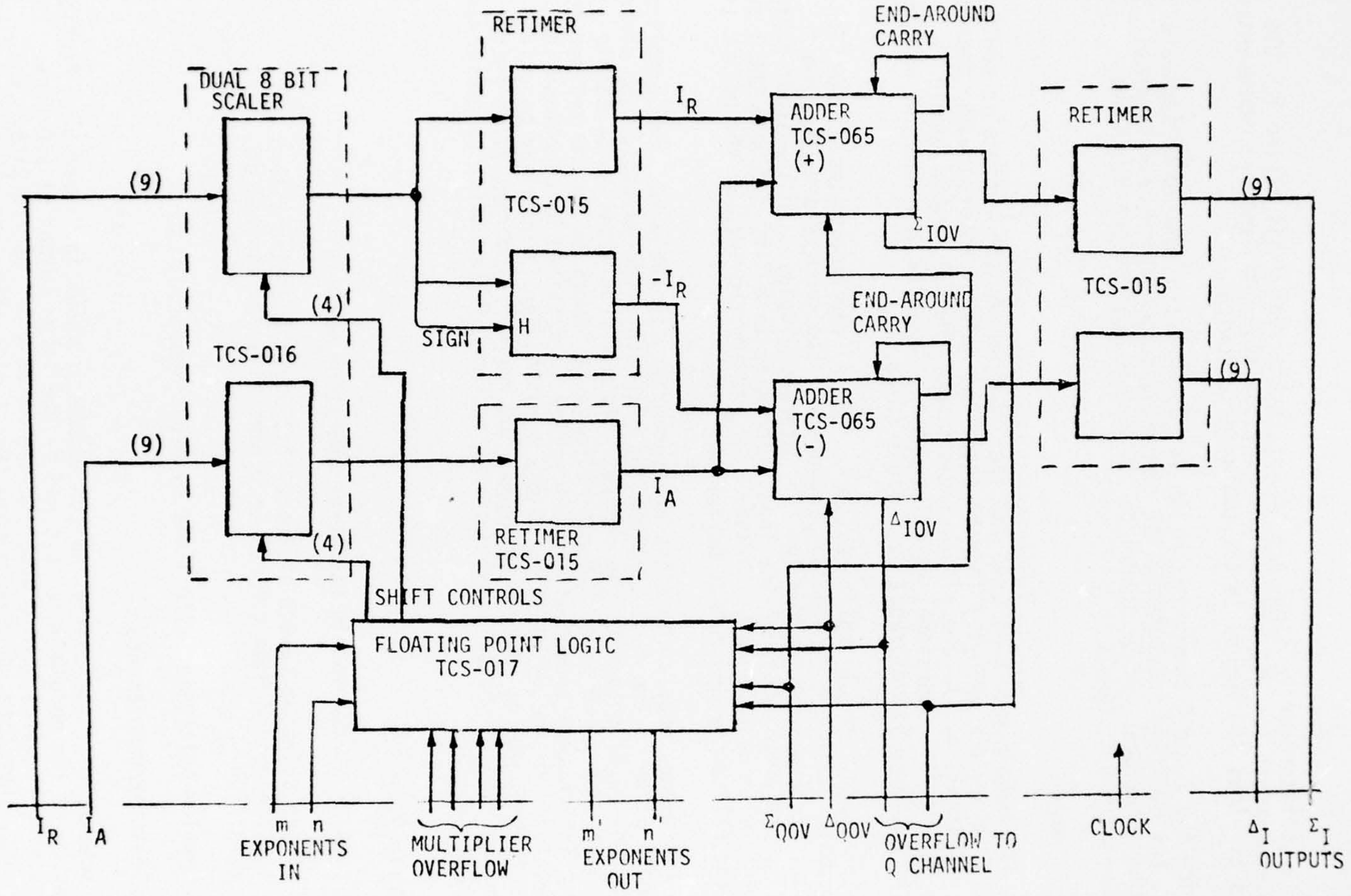


FIGURE 48. ADDER/SUBTRACTOR MODULE

multiplier rotator) or I_A input are scaled in the dual 8-bit scaler TCS-016. The scaler outputs are retimed and both the positive and negative representation of I_R is provided for inputs to the two adder circuits.

Since the I and Q components of an output word are located on separate modules, the overflow interchange for keeping a common exponent must cross the module interface. Other inputs to the module include the clock and multiplier overflow signals.

6.1.3 FFT Memory

The FFT memory module is used for the input and output buffers in addition to the FFT interstage delays. In each case, the delays are programmed depending on mode and location. A general block diagram of the FFT memory module is shown in Figure 49. Functional descriptions of the three applications of the module can be found in reference (2), pages 60-68.

Up to 11 programmable shift registers (GUA's) can be mounted on the FFT memory module. With the module fully populated, two modules are required to provide the 22 bits per word for the FFT interstage delay. In the input buffer, eight GUA's are required since the input word size is 8 bits for both the I and Q components with no exponents. Driver circuits are provided on the module to control the FFT memory and output buffer switches.

The FFT memory module functions are summarized in Table 28. One complication to the application is the fact that the GUA gives inverted outputs. The correct sign/magnitude (S/M) output to the complex multiplier is obtained by complementing in the GUA which gives $\overline{S/M}$ out followed by inversion in the retimer to give S/M. The delayed output of the FFT memory is obtained by complementing in the retimer. The delay function of the input buffer is obtained by using the discretionary wiring package to bypass the retimer. Correct delay increments are obtained in the switched register section of the buffer by crossover of the delay sets.

6.1.4 Control Switch

The control switch module has three operating modes:

1. Simple selector switch of two 22 bit words operated either statically for mode control or dynamically for control of data flow.

2. Complex multiplier with 1 bit by 8 bit inputs.

3. Selector/Complementer.

The schematic of the control switch module is shown in Figure 50. Particulars of the design of the module were governed by the large number of I/O pins. The basic switching requirement dictates the use of 22 x 3 = 66 I/O pins. The controls must, therefore, occupy no more than 2 pins after the allotment of 10 grounds and 2 power pins. The control inputs have been minimized by using a discretionary wiring package to enable the module to accommodate the three modes.

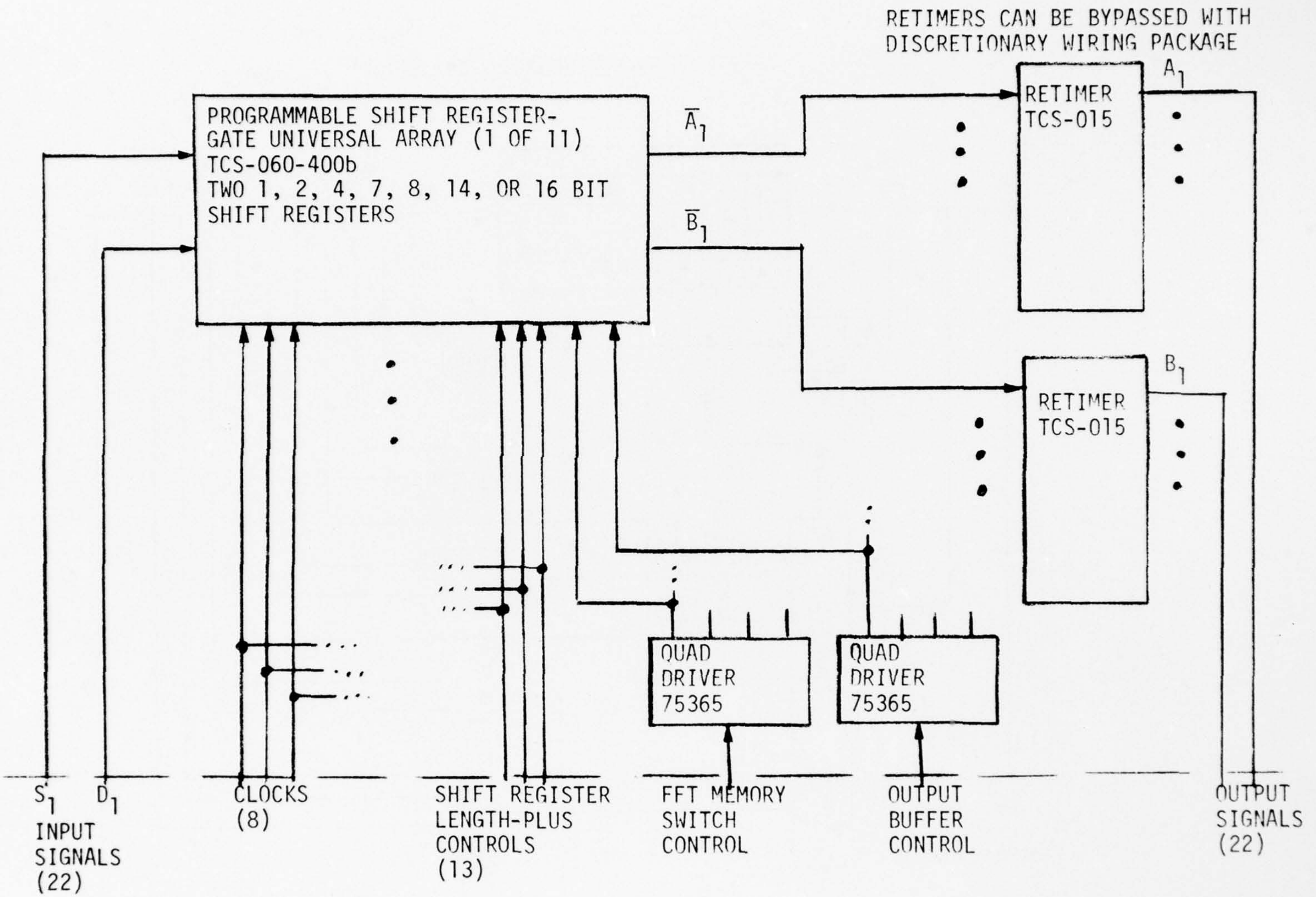
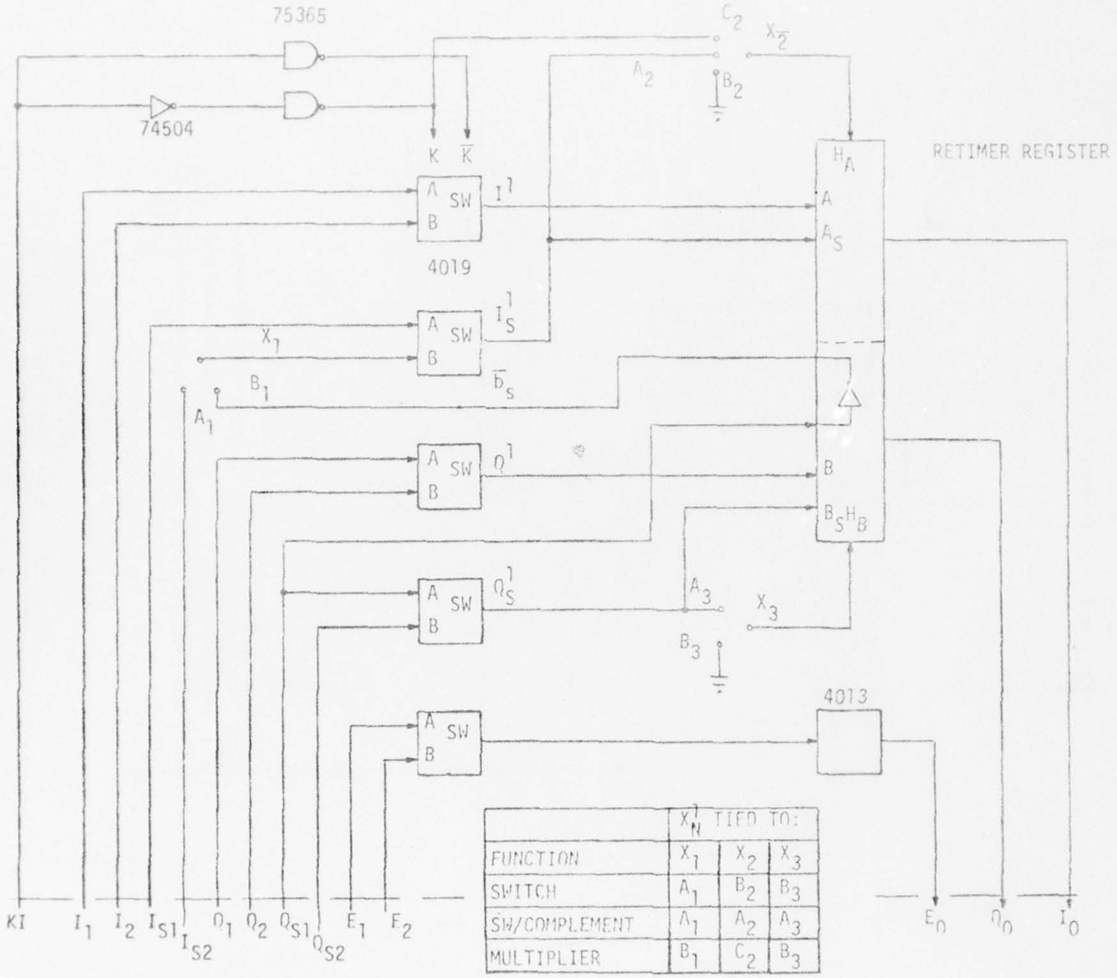


FIGURE 49. FFT MEMORY MODULE



	X^1_N TIED TO:		
FUNCTION	X_1	X_2	X_3
SWITCH	A_1	B_2	B_3
SW/COMPLEMENT	A_1	A_2	A_3
MULTIPLIER	B_1	C_2	B_3

FIGURE 50. CONTROL-SWITCH MODULE

TABLE 28
FFT MEMORY MODULE FUNCTIONS

FUNCTION	REQUIREMENT	OBTAINED BY
FFT Memory	S/M Output	Complement in GUA Gives $\overline{S/M}$ Followed by Inversion in Retimer
	1's Comp. Output Delayed	Complement in Retimer
Input Buffer	Delay (32,16,8) (Sample Clock)	Bypass Retimer Cascade Delay Sets (4,6,16) & (4,10,16)
	Switched Buffer S/M Output (Sample/Process Clock)	Cascade Delay Sets with Crossover Pattern
Output Buffer	Switched Buffer (Sample/Process Clock)	Normal Operation with or without Retimer

In the selector switch configuration, which always holds for the exponent bits, the complement controls of the retimers are tied to ground and the inputs go to their normal switch locations. When operating as a complementer, the sign inputs are tied to the retimer complement controls.

Operation as a 1 x 8 bit complex multiplier functions by shifting the phase angle 0° or 90°. The two desired output conditions are:

$0° : (1 + j0)(a + jb) = a + jb$
$90°: (0 + j1)(a + jb) = -b + ja$

Therefore, for the 90° shift, the imaginary output is selected from the real input and the negative of the imaginary input is fed to the output. Negating the imaginary input requires not only complementing the data, but also inverting the sign bit. The sign inversion is accomplished by a discretionary wiring of the sign of the imaginary input to the real channel switch.

6.1.5 Level Translator

The level translator module shown functionally in Figure 51 provides TTL to CMOS/SOS interfacing for the PWP control functions. The data is re-clocked once on the module in hex-D flip-flops (74S174), is level shifted with 75365 TTL to CMOS drivers and is reclocked at the CMOS/SOS levels with the TCS-015 retimer.

6.1.6 Reorder Memory

The reorder memory module is a 11 bit x 1024 word random access memory. The basic design of the module is straight-forward as shown in Figure 52. All of the mode and

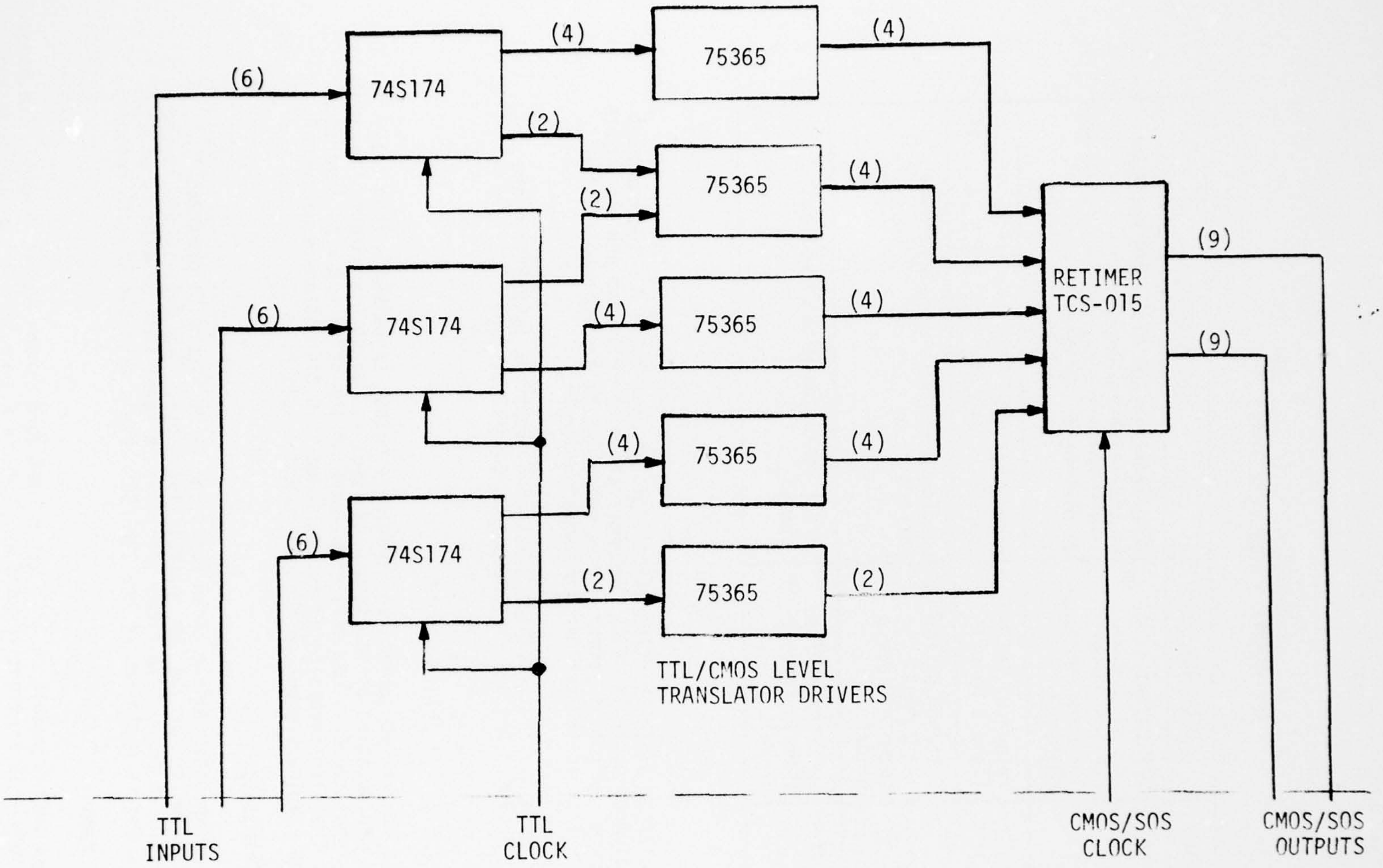


FIGURE 51. LEVEL TRANSLATOR MODULE FUNCTION

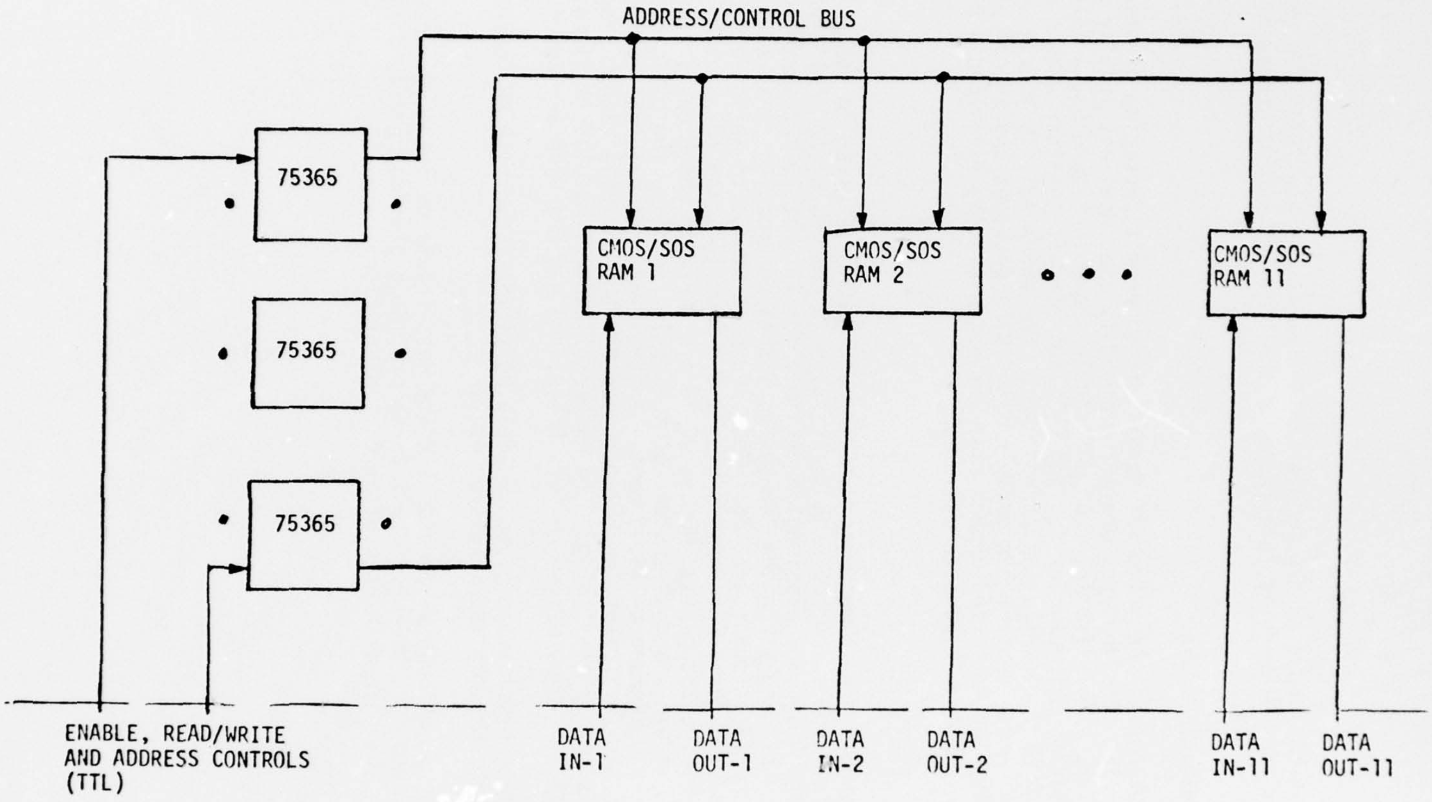


FIGURE 52. REORDER MEMORY MODULE FUNCTION

address controls input to the module at TTL levels and use 75365 TTL CMOS/SOS level translator-drivers as interfaces to the CMOS/SOS RAM's. Three driver circuits are used.

6.1.7 Universal Modules

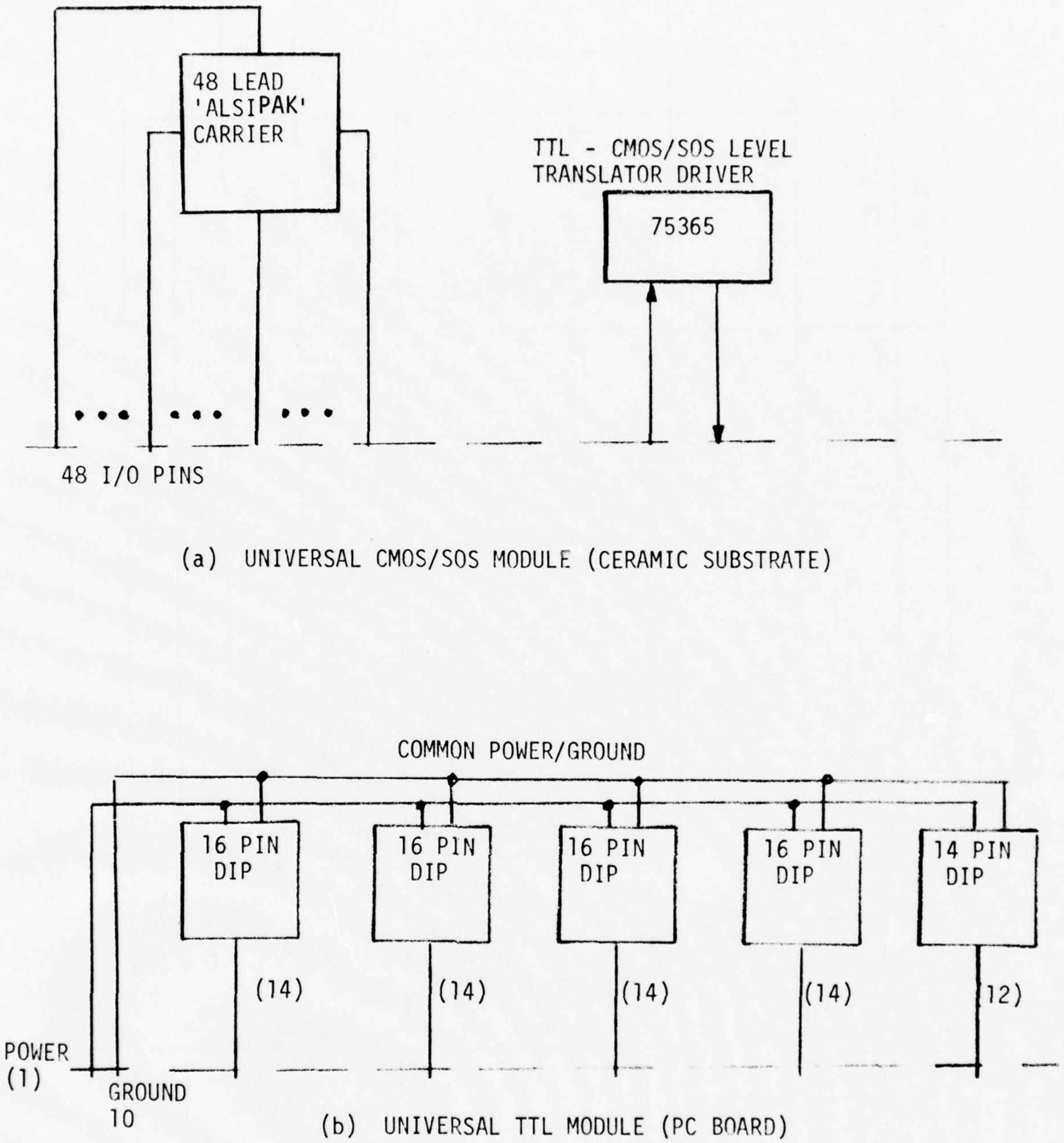
The purpose of the universal modules is to permit packaging of discrete CMOS/SOS or TTL IC's in cases where the quantity used does not justify a separate module design. The universal CMOS/SOS module shown in Figure 53(a) holds one 48 pin 'ALSIPAK' chip carrier for any CMOS/SOS circuit together with a wired-in 75365 TTL CMOS/SOS level translator driver. The universal TTL module shown in Figure 53(b) holds four 16 pin dual-in line (DIP) packages and one 14 pin DIP package. The ten dedicated ground pins prevented placing five 16 pin packages on the module with all pins accessible.

6.2 MODULE SUMMARIES

A listing of the components used on each module is given in Table 29. Table 30 lists the number of each module type in the system and FFT's only together with the power dissipation estimate for 15 volt 10 MHz operation. The total number of modules in the PWP system increased from 154 in an earlier estimate to 192 in the final configuration. This increase is due to an increase of 20 TTL modules used for clock drivers and 18 TTL modules added for various control functions. Since the maximum number of 16 pin circuits which could be mounted on each TTL module was four instead of the estimated five, the efficiency of these modules decreased by 25 percent. In addition, the original estimate did not include sufficient control circuits for the variable length programming of the FFT's and system timing. The number of control modules is divided among the reorder memory, I/O buffer, FFT's, and various PROM storage functions.

The power dissipation given in Table 30 is for 15 volt 10 MHz operation and represents the maximum expected level.

Figures 54 through 61 provide summary data sheets on the eight module types developed for the PWP. The figures provide a physical and functional description together with electrical characteristics.



(a) UNIVERSAL CMOS/SOS MODULE (CERAMIC SUBSTRATE)

(b) UNIVERSAL TTL MODULE (PC BOARD)

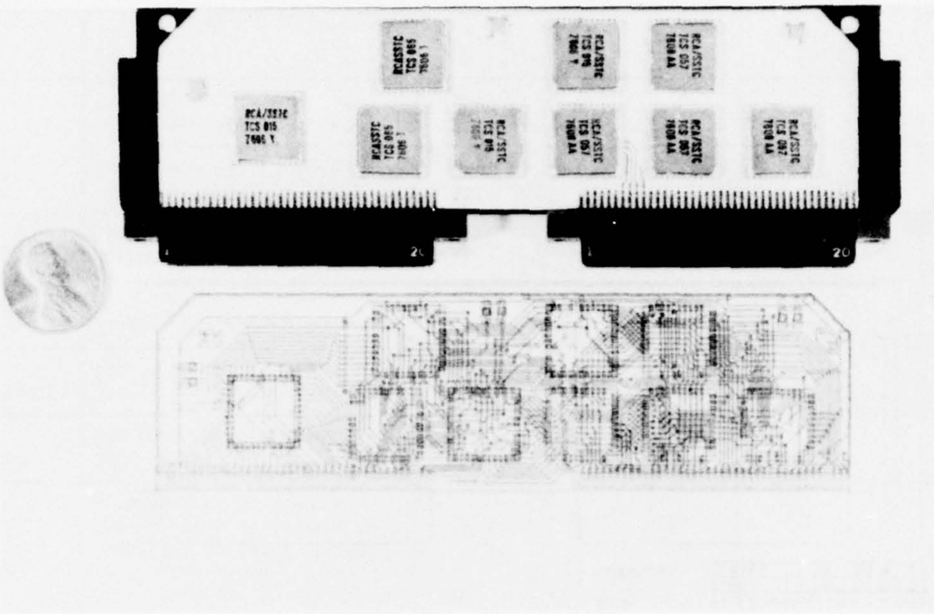
FIGURE 53. UNIVERSAL MODULE FUNCTIONS

TABLE 29
MODULE COMPONENT TABULATION

MODULE	FUNCTION	TECHNOLOGY	QUANTITY
Complex Multiplier	° 8 Bit Multiplier Chip ° Dual 9 Bit Retimer Register ° 8 Bit Adders	All CMOS/SOS	4 3 2
Complex Adder/ Subtractor	° Dual 8 Bit Scaler ° Dual 9 Bit Retimer Register ° Floating Point Logic Array ° 8 Bit Adder	All CMOS/SOS	1 3 1 2
FFT Memory (Programmable Delay)	° Gate Universal Array Programmable Shift Register ° Dual 9 Bit Retimer Register ° TTL-MOS Level Translator/Driver	CMOS/SOS CMOS/SOS TTL	11 2 2
Control Switch	° Quad 2-1 Data Selectors ° Dual D Flip Flop ° Dual 9 Bit Retimer Register ° Exclusive OR Gate ° TTL-MOS Level Translator/Driver ° Discretionary Wiring Package	CMOS CMOS CMOS/SOS CMOS TTL -	6 2 1 1 1 1
Level Translator	° Hex D Flip Flop ° Dual 9 Bit Retimer Register ° TTL-MOS Level Translator/Driver (75365)	TTL CMOS/SOS TTL - CMOS/SOS	3 1 5
Reorder Memory (1024 x 11 Bits)	° 1024 x 1 Random Access Memory ° TTL-MOS Level Translator/Driver	CMOS/SOS TTL	11 3
Universal SOS	° 1 CMOS/SOS LSI Circuit, One Driver (75365)	CMOS/SOS TTL	1 1
Universal TTL	° Up to Four 16 Pin and one 14 Pin TTL Circuits	TTL	5

TABLE 30
MODULE SUMMARY FOR PWP SYSTEM AND FFT'S

MODULE	FUNCTION	PWP	FFT'S ONLY	MODULE POWER DISSIPATION (10MHz,15V)	PWP TOTAL POWER (10MHz,15V)	FFT'S POWER (10MHz, 15V)
Complex Multiplier	8 Bit Plus Sign Multiplication of Two Complex I and Q Data Words	13	8	2.27	29.5	18.2
Complex Adder/ Subtractor	Addition and Subtraction of 8-Bit Plus Sign Words With Floating Point Characteristic. Includes Input and *Output Scaling. (2 Required* for Full Complex Addition.)	24	24	1.49	35.7	35.7
FFT Memory (Programmable Delay)	Shift Register Memory for up to 6 Stage Radix - 2 FFT. Adaptable to Input and Output Buffering Requirement of Step Transform Process.	38	20	3.89	147.82	77.8
Control Switch	Data Selector, 1 Bit x 8 Bit Complex Multiplier	15	8	2.15	32.25	17.2
Level Translator	Shifts and Reclocks 18 TTL Inputs to SOS Levels	13	8	2.85	37.05	22.8
Reorder Memory (1024x11 Bits)	11 Bits x 256 Word Memory	24	0	2.68	64.32	0
Universal CMOS/SOS	1 CMOS/SOS LSI Circuit, 1 Level Translator Driver	11	6	.5	5.5	3.0
Universal TTL	Four 16-Pin TTL Circuits, One 14-Pin TTL Circuit	34	14	1.25	42.5	17.5
	Dedicated CMOS/SOS Clock Drivers	20	10	2.6	52.0	26.0
	TOTALS	192	98	-	446.64	218.2



GENERAL

The Complex Multiplier Module performs the function (a+jb)(c+jd)=(e+jf) where a, b, c and d are 8 bit plus sign, sign-magnitude numbers; and e and f are 8 bit plus sign, one's complement numbers. The outputs are rounded off. Should there be an overflow in the complex addition, both real and imaginary parts are shifted and an indication of overflow is given.

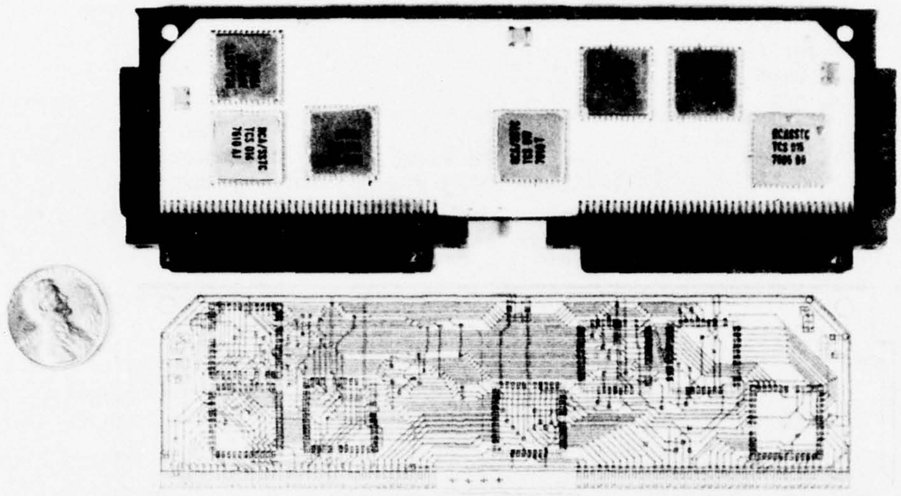
PHYSICAL DESCRIPTION

Active Devices
  3 - TCS015 CMOS/SOS Dual 9 Stage Retimer (48 Pin Leadless Carrier)
  4 - TCS057 CMOS/SOS 8x8 Bit Multiplier (48 Pin Leadless Carrier)
  2 - TCS065 CMOS/SOS 8 Bit Digital Adder (48 Pin Leadless Carrier)
Equivalent Gates/Module - 3545

ELECTRICAL DESCRIPTION

Supply Voltage
  VDD - 3V to 15V
Power Supply Current Drain
  IDD = 155mA @ VDD = 15V and f = 10 MHz
Operating Speed - 0 to 10 MHz
Reclockings Needed to Complete Function - 2

FIGURE 54. COMPLEX MULTIPLIER MODULE DATA



GENERAL

This module takes the sum and difference of two one's complement eight bit plus sign numbers with four bit exponents. The two input numbers are shifted to produce the same exponents. The resultant numbers, A and B, are then added and subtracted from each other to produce A+B and A-B. Should there be an overflow in either addition, the exponent of that addition is incremented and the outputs shifted. When used in pairs, these modules can be configured to perform complex arithmetic.

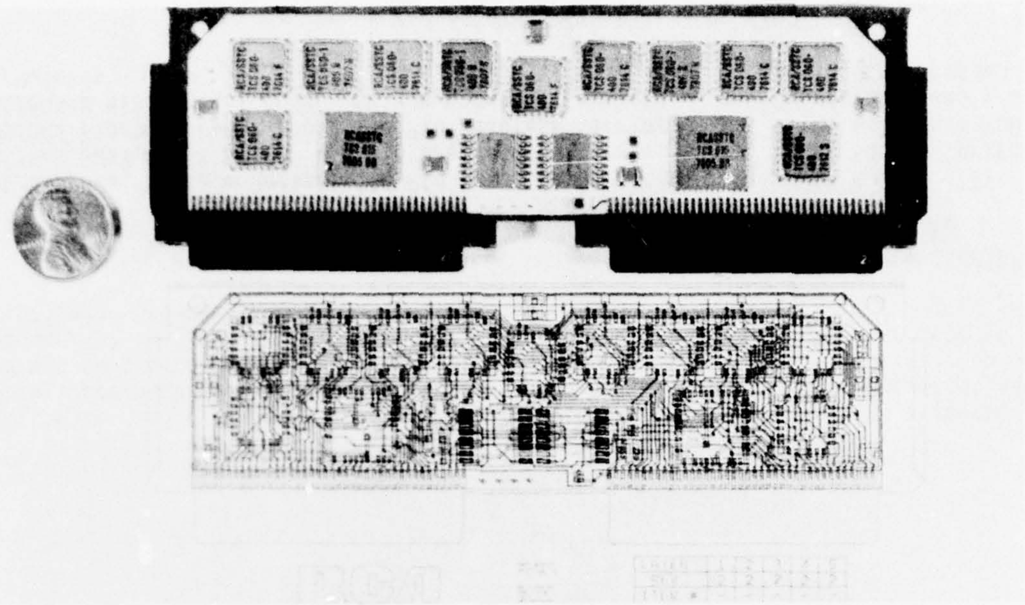
PHYSICAL DESCRIPTION

Active Devices
  3 - TCS015 CMOS/SOS Dual 9 Stage Retimer (48 Pin Leadless Carrier)
  1 - TCS016 CMOS/SOS Dual 8 Bit Scaler (48 Pin Leadless Carrier)
  1 - TCS017 CMOS/SOS Floating Point Logic (48 Pin Leadless Carrier)
  2 - TCS065 CMOS/SOS 8 Bit Digital Adder (48 Pin Leadless Carrier)
Equivalent Gates/Modules - 1150

ELECTRICAL DESCRIPTION

Supply Voltage
  VDD - 3V to 15V
Power Supply Current Drain
  IDD = 100mA @ VDD = 15V and f = 10 MHz
Operating Speed - 0 to 10 MHz
Reclockings Needed to Complete Function - 2

FIGURE 55. COMPLEX ADDER/SUBTRACTOR MODULE DATA



GENERAL

The Programmable Delay Module is an 11 bit programmable shift register. Each bit has two channels with switching that can be used to implement delays of zero to 46 clock pulses. There are controls to complement the outputs with two of the eleven bits being independent from the rest. All of the data bits are reclocked before leaving the module.

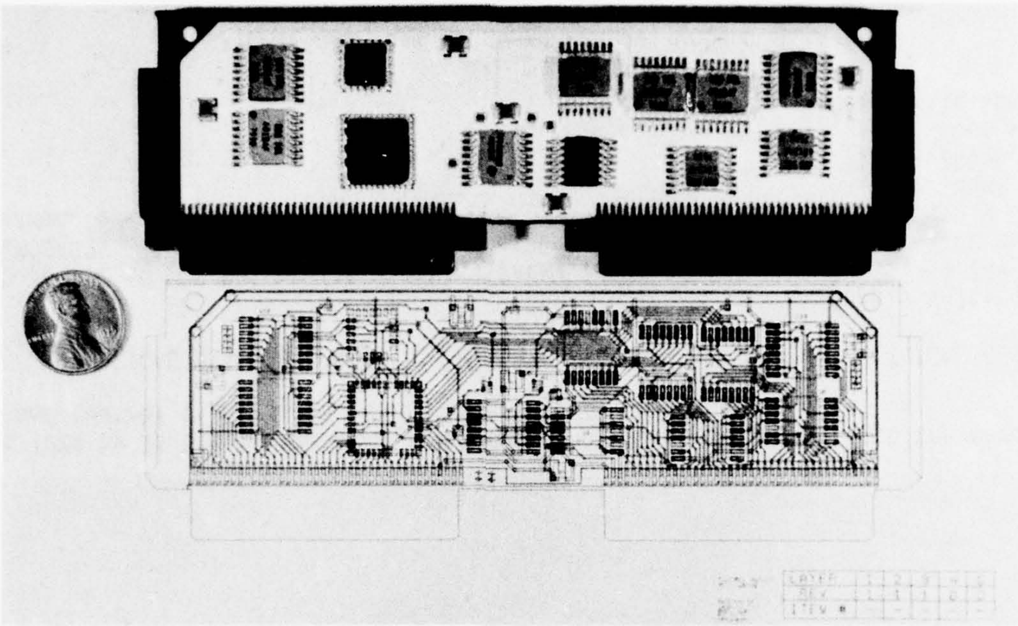
PHYSICAL DESCRIPTION

Active Devices
  2 - 75365 Quad TTL to MOS Driver (16 Pin Flat Pack)
  2 - TCS015 CMOS/SOS Dual 9 Stage Retimer (48 Pin Leadless Carrier)
  11 - TCS060 400B CMOS/SOS Programmable Length Shift Register
    (28 Pin Leadless Carrier)
Equivalent Gates/Modules - 2106

ELECTRICAL DESCRIPTION

Supply Voltage
  VCC - 4.75V to 5.25V
  VDD - VCC to 15V
Power Supply Current Drain
  ICC = 37mA @ VCC = 5.25V
  IDD = 225mA @ VDD = 15V and f = 10 MHz
Operating Speed - 0 to 10 MHz
Reclockings Needed to Complete Function -
    Programmable from 1 to 47.

FIGURE 56. PROGRAMMABLE DELAY (FFT MEMORY) MODULE DATA



GENERAL

The Control Switch Module can be configured to perform one of three functions. By means of jumpers within a 24 pin leadless carrier mounted on the module, the module can: 1) select one of two 22 bit words; 2) select one of two 22 bit words and perform the complement of the word; 3) perform the complex multiplication (a+jb)(1+j0) = (a+jb) or (a+jb)(0+j) = -b+ja.

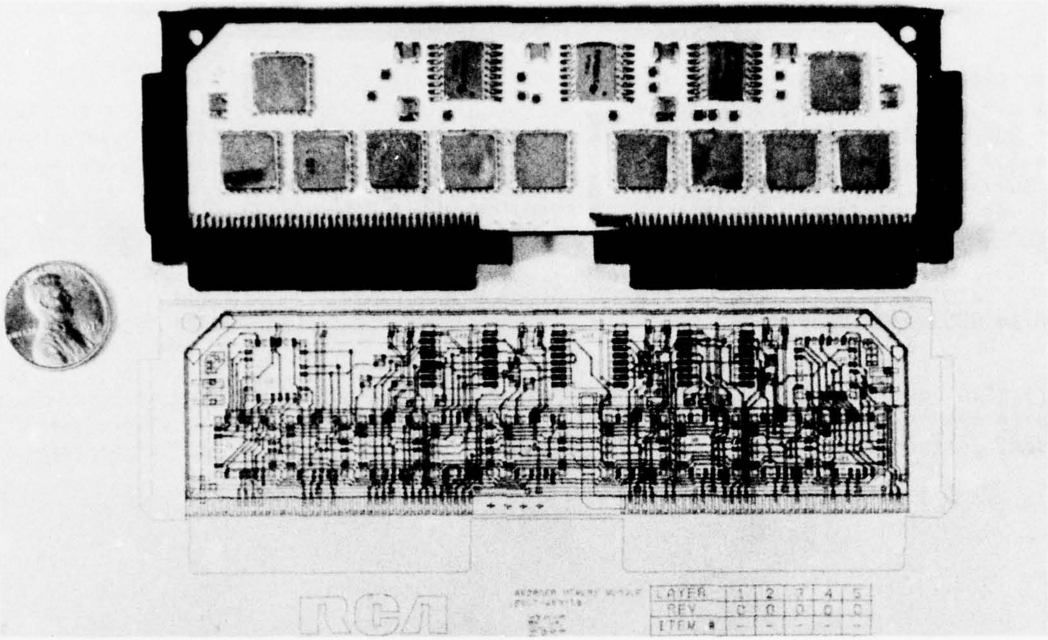
PHYSICAL DESCRIPTION

Active Devices
  2 - 4013   CMOS Dual D Flip-Flow (14 Pin Flat Pack)
  6 - 4019   CMOS Quad AND-OR Select Gate (16 Pin Flat Pack)
  1 - 74S04   Schottky TTL Hex Inverters (14 Pin Flat Pack)
  1 - 75365   Quad TTL to MOS Driver (16 Pin Flat Pack)
  1 - TCS015 CMOS/SOS Dual 9 Stage Retimer (48 Pin Leadless Carrier)
Equivalent Gates/Module - 219

ELECTRICAL DESCRIPTION

Supply Voltage
  VCC - 4.75V to 5.25V
  VDD - VCC to 15V
Power Supply Current Drain
  ICC - 50mA @ VCC = 5.25V
  IDD - 185mA @ VDD - 15V and f = 10 MHz
Operating Speed - 0 to 10 MHz
Reclockings Needed to Complete Function - 1

FIGURE 57. CONTROL SWITCH MODULE DATA



GENERAL

This module is a 1024 by 11 bit random access memory. The inputs and outputs to the memory are at CMOS levels and the address and control signals are at TTL levels.

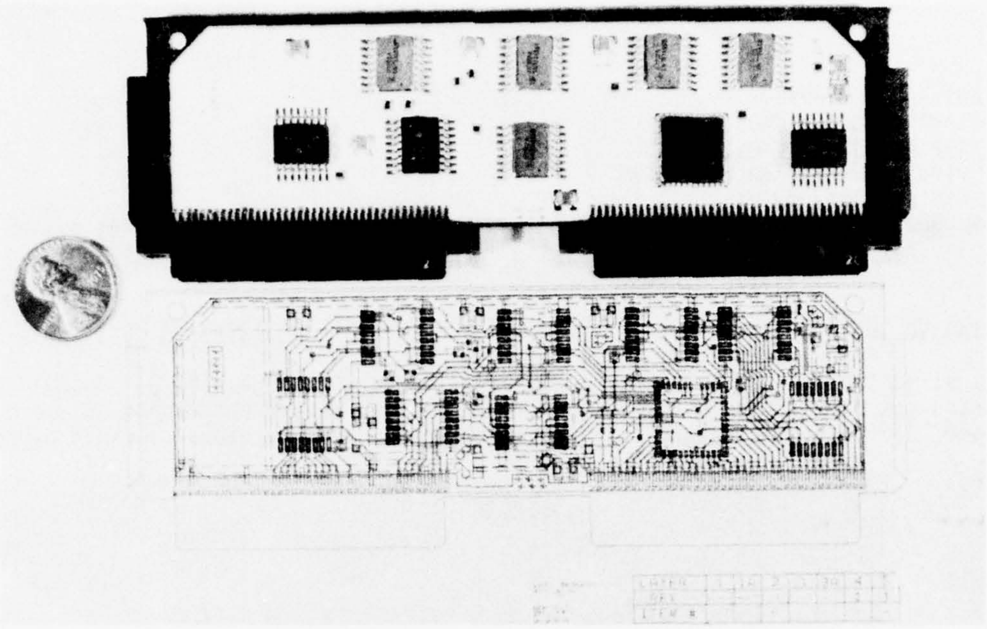
PHYSICAL DESCRIPTION

Active Devices
  3 - 75365   Quad TTL to MOS Driver (16 Pin Flat Pack)
  11 - TA6780 CMOS/SOS - 1024X1 RAM (28 Pin Leadless Carrier)
Equivalent Gates/Module - 27512

ELECTRICAL DESCRIPTION

Supply Voltage
  VCC - 4.75V to 5.25V
  VDD - VCC to 15V
Power Supply Current Drain
  ICC = 85mA @ VCC = 5.25V
  IDD = 245mA @ VDD = 15V and memory read sequentially at f = 9 MHz.
Read/Write Speed - 110 ns

FIGURE 58. RAM (REORDER MEMORY) MODULE DATA



GENERAL

This module provides a TTL to CMOS level translation for 18 input bits. The input, at a TTL level, is reclocked, level translated to CMOS levels, and reclocked again. Six of the inputs have their reclocked signal available as outputs at a TTL level.

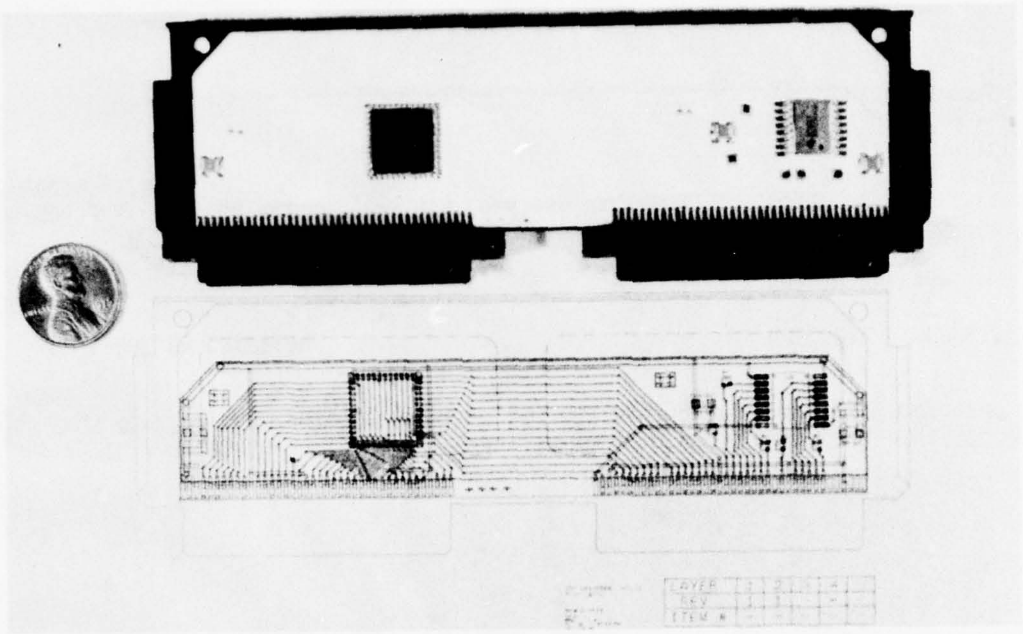
PHYSICAL DESCRIPTION

Active Devices
  3 - 74LS174 TTL Hex D Flip-Flop (16 Pin Flat Pack)
  5 - 75365 Quad TTL to MOS Driver (16 Pin Flat Pack)
  1 - TCS015 CMOS/SOS Dual 9 Stage Retimer (48 Pin Leadless Carrier)
Equivalent Gates/Module - 217

ELECTRICAL DESCRIPTION

Supply Voltage
  VCC - 4.75V to 5.25V
  VDD - VCC to 15V
Power Supply Current Drain
  ICC = 205mA @ VCC = 5.25V
  IDD - 115mA @ VCC = 15V and f = 10 MHz
Operating Speed - 0 to 10 MHz
Reclockings Needed to Complete Function - 2

FIGURE 59. LEVEL TRANSLATOR MODULE DATA



GENERAL

This universal module provides capabilities of mounting any device packaged in a 48 pin leadless carrier and having all pins available at the *module connector. The module* also contains a quad TTL to MOS driver with all outputs damped to control ringing. A reference high to pull-up unused TTL inputs on the driver is also provided.

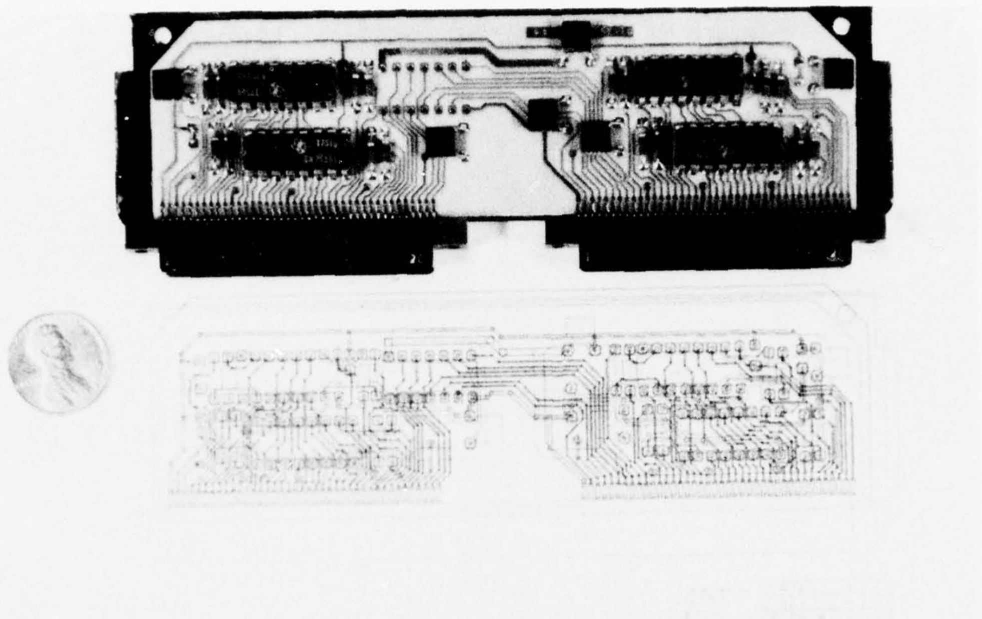
PHYSICAL DESCRIPTION

Active Devices
  1 - 75365 Quad TTL to MOS Driver (16 Pin Flat Pack)
  1 - Any active device in a 48 Pin Leadless Carrier

ELECTRICAL DESCRIPTION

Supply Voltage
  VCC - 4.75V to 5.25V
  VDD - VCC to 15V (*Dependent on VDD max of device* in 48 Pin Leadless Carrier).
Power Supply Current Drain
  ICC = 18mA @ VCC = 5.25V
  IDD = 50mA + IDD of device in 48 Pin Leadless Carrier @ VCC = 15V

FIGURE 60. UNIVERSAL CMOS/SOS MODULE DATA



GENERAL

The Universal TTL Module has locations available to mount four 16 pin DIP's and one 14 pin DIP. All of the pins are brought out to the connector with the exception of power and ground which are dedicated on the devices to conform with standard TTL. In the 16 pin DIP locations, 75365-TTL to MOS Drivers may be mounted with locations for damping resistors provided on the module. Through the use of jumpers on the module, 14 pin DIP's may be used in the 16 pin DIP locations.

PHYSICAL DESCRIPTION

Active Devices
  1 - Device in a 14 pin DIP
  4 - Device in 16 pin DIP's

ELECTRICAL DESCRIPTION

Supply Voltage
  VCC - 4.75V to 5.25V
  VDD - VCC to 15V when 75365-TTL to MOS Drivers are used.
Power Supply Current Drain - Dependent on current drain of devices used.

FIGURE 61. UNIVERSAL TTL MODULE DATA

PrecedING Page BLank - NOT FILMed

SECTION VII

PWP CONTROL SYSTEM

7.1 OVERALL DESCRIPTION

The PWP uses pipeline architecture to implement the FFT and maintain the data rate. This architecture creates complex control system problems when mode changes are required and delays are dropped from or added to the pipeline. This is because new sets of control signals are required and must be resynchronized to a new pipeline of larger or smaller size.

A block diagram of the PWP pipeline is shown in Figure 62. The first two stages of the forward FFT can be bypassed and the last two stages of the inverse FFT can be bypassed. When a mode change is made, one or more of these stages are bypassed. Not only does this affect the control signal timing, but new data sets must be generated in the sin/cos reference generators and the deramping and phase correction memories. These data sets must also be synchronized with the data window.

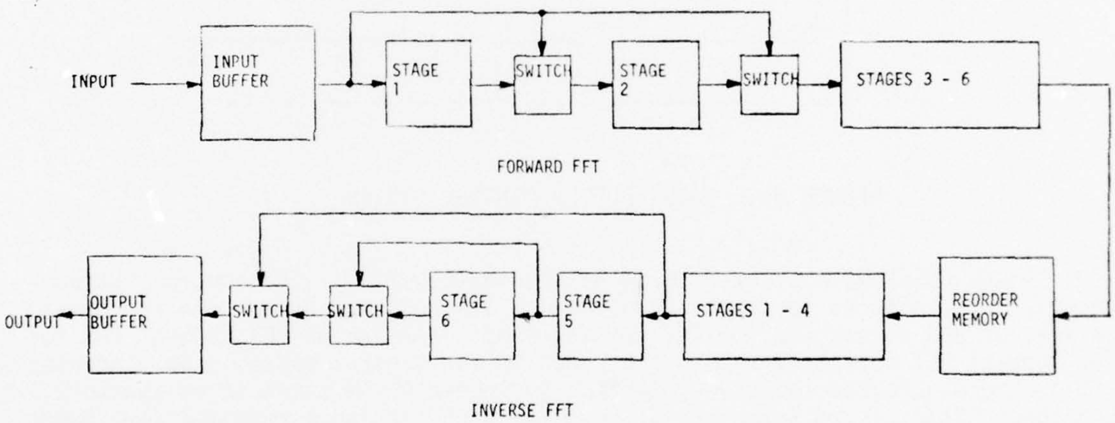


FIGURE 62. PWP PIPELINE

Several approaches can be taken in the control system design. However, it has been determined that significant savings in the number of circuit components as well as more efficient operation can be obtained by using Programmable Read Only Memories (PROM's) to store many of the control functions. Those control functions which change when a mode change is effected are stored in PROM's. In addition to control functions, all the ramping, sin/cos references, phase correction, and amplitude correction information is stored in PROM's.

The control systems for the input and output buffers, forward and inverse FFT's, and reorder memories are separate,disconnected control systems which are synchronized by means of a sync bus. There is an address link between the input buffer control system and the forward FFT control system.

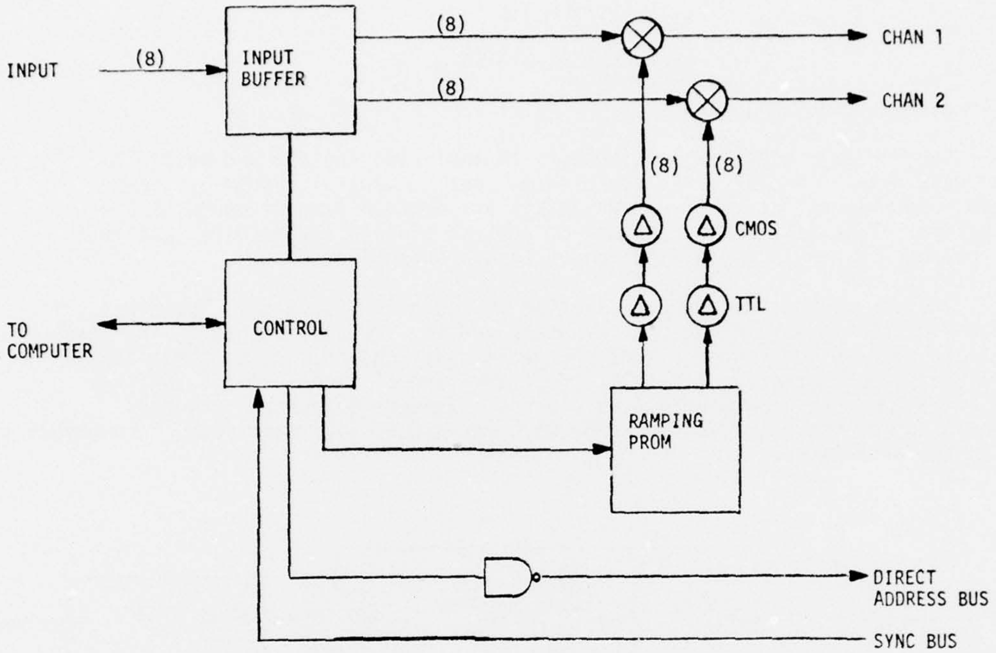


FIGURE 63. INPUT BUFFER CONTROL SYSTEM

Figure 63 shows a block diagram of the input buffer. The control system generates the address for the control PROM's in the input buffer control system as well as the addressing for the ramping PROM's and the DIRECT ADDRESS BUS for the forward FFT control system. The input buffer control system also generates timing signals to the computer interface to indicate the start of an aperture. The input buffer is an interface between the 8.75 MHz input rate and the 10 MHz process rate and requires that the clock to the registers in the buffer be switched from 8.75 MHz to 10 MHz. The input buffer control system performs this function also. The ramping multipliers are considered to be a part of the input buffer and so the PROM's which contain the frequency ramps and input weighting are considered part of the control system. They are addressed directly from the control counter. The circuit elements marked Δ represent digital delays in the level translator module.

A block diagram of the forward FFT is shown in Figure 64. The control signals into the FFT are the sin/cos references and the sign/switch controls. The sin/cos references are stored in PROM's all of which are identical. The addressing and timing for the reference PROM's is contained in the INDIRECT ADDRESS PROM. When the mode changes to change the length of the FFT, a different area of the INDIRECT ADDRESS PROM is accessed. The address from the DIRECT ADDRESS BUS then cycles through the proper address sequence for the INDIRECT

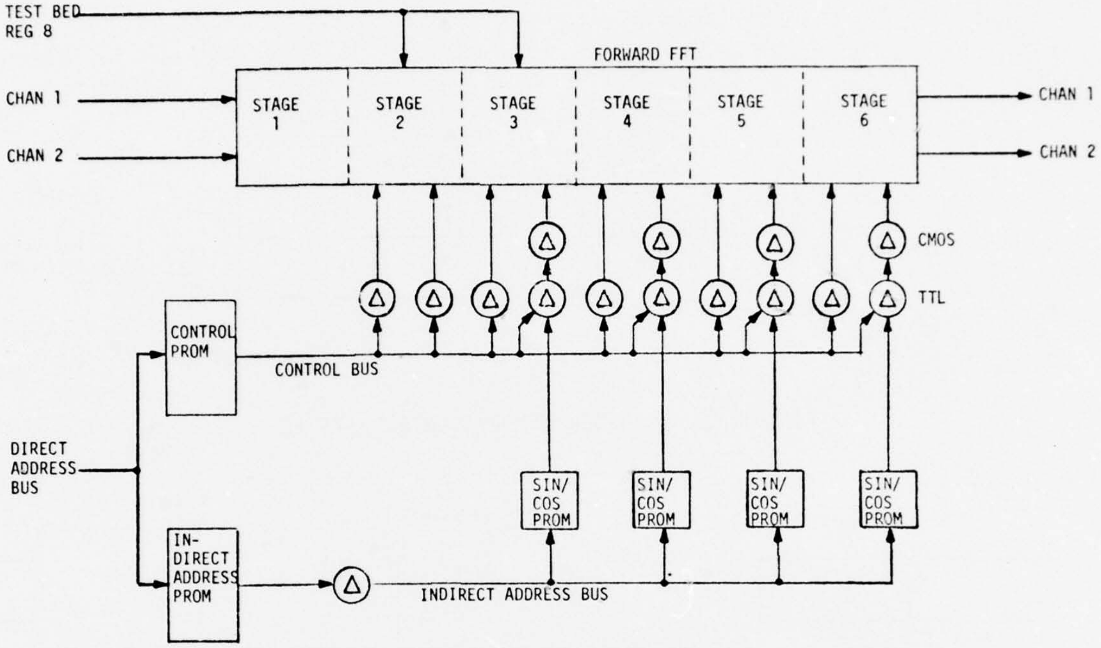


FIGURE 64. FORWARD FFT CONTROL SYSTEM

ADDRESS BUS which in turn accesses the proper sin/cos reference. The change in timing encountered by changing modes is accomplished by rotating the address sequence in that section of the INDIRECT ADDRESS PROM accessed by the mode control bits. The switch bits and sign bits for the cos reference are contained in the CONTROL PROM and transmitted along the CONTROL BUS. The switch bits control the switching in the FFT memories. Several static controls such as stage delay codes for the FFT memories are hard wired at the module inputs.

A block diagram of the reorder memory is shown in Figure 65. The reorder memory control system controls not only the reorder memory but also the sync generator. The reorder memory control system contains a counter which spans the entire length of an input waveform. Since this is the largest counter in the system, it makes sense that it should be the basis for the sync generator. In this case, the sync generator is nothing more than a PROM which has a single bit programmed to sync the counters for the rest of the system. The timing is accomplished by rotating the pulse in the memory. The oscillator is also contained in the reorder memory control system.

A block diagram of the inverse FFT is shown in Figure 66. The control system is almost like that of the forward FFT. The most notable exception is the fact that the sin/cos reference PROM's are addressed from the DIRECT ADDRESS BUS. The reason for this is that neither the timing nor the reference sequence are altered by a mode change because the output data is tapped off the pipeline by a switch at the output. The inverse FFT control system also includes the

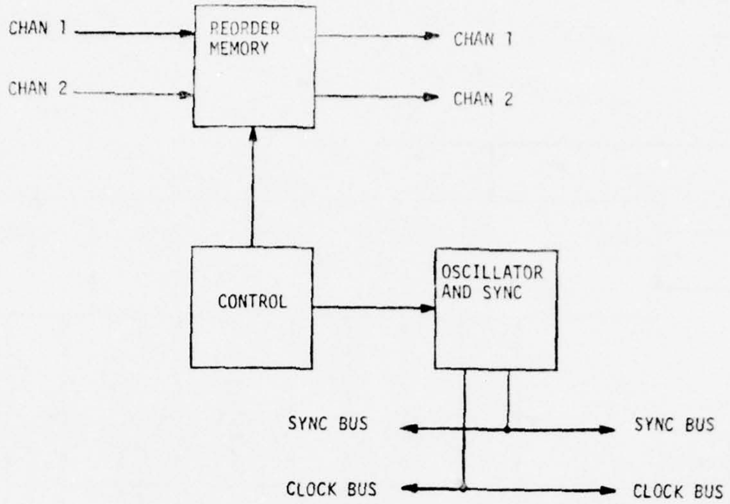


FIGURE 65. REORDER MEMORY CONTROL SYSTEM

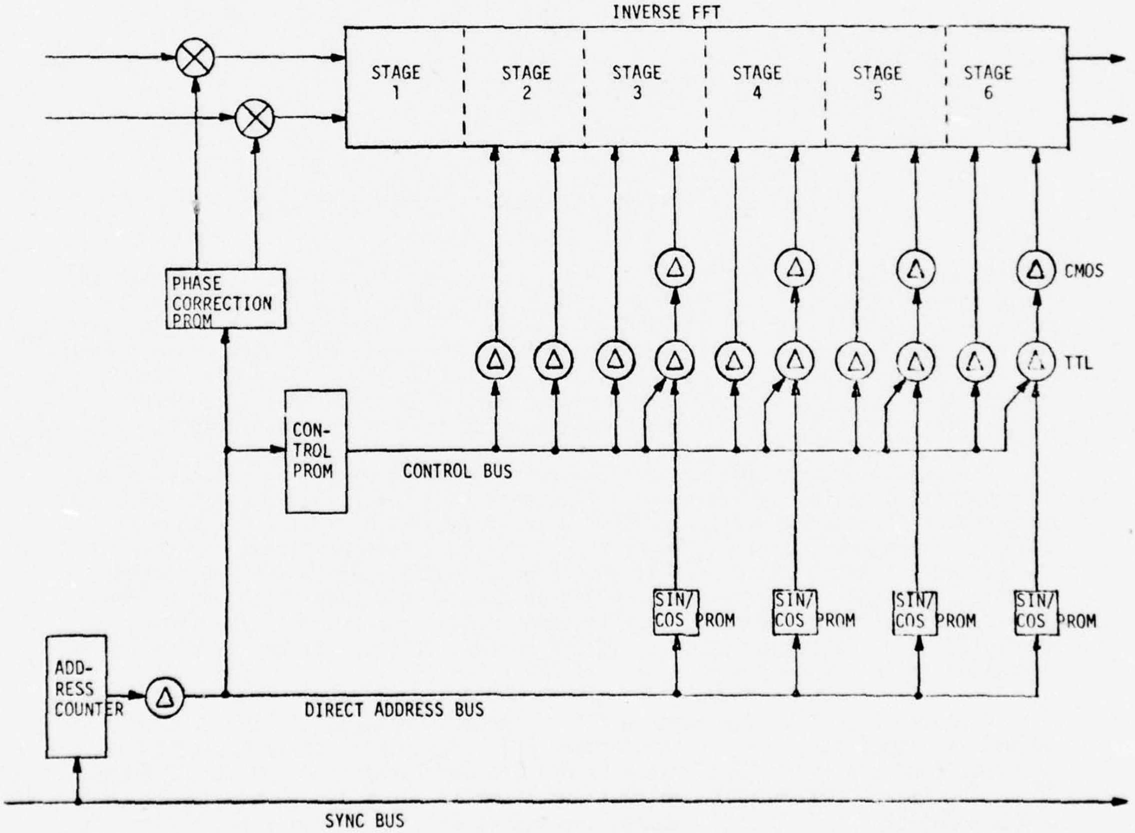


FIGURE 66. INVERSE FFT CONTROL SYSTEM

PHASE CORRECTION PROMS at the beginning of the FFT.

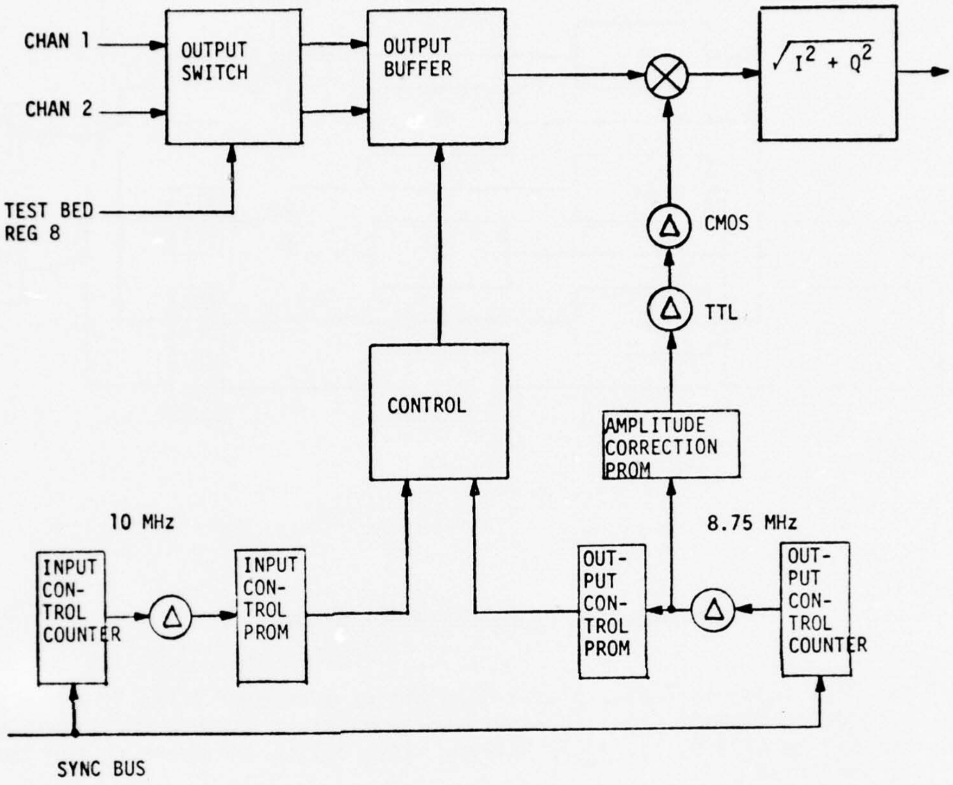


FIGURE 67. OUTPUT BUFFER CONTROL SYSTEM

A block diagram of the output buffer is shown in Figure 67. The output buffer function is to discard the redundant data processed due to the overlaps at the input buffer. This reduces the data output rate from the 10 MHz process rate to the 8.75 MHz sample rate. A dual set of controls are required for this. One set operates at 10 MHz and the other at 8.75 MHz. They are synchronized by the sync bus. The input control system sets the switching time for the memories and the clock. The AMPLITUDE CORRECTION PROM is addressed by the output control counter.

7.2 INPUT BUFFER

The purpose of the input buffer is to provide the overlaps in data input apertures required for the PWP algorithm, and to split the data for the radix-2 operation. A block diagram of the input buffer is shown in Figure 68.

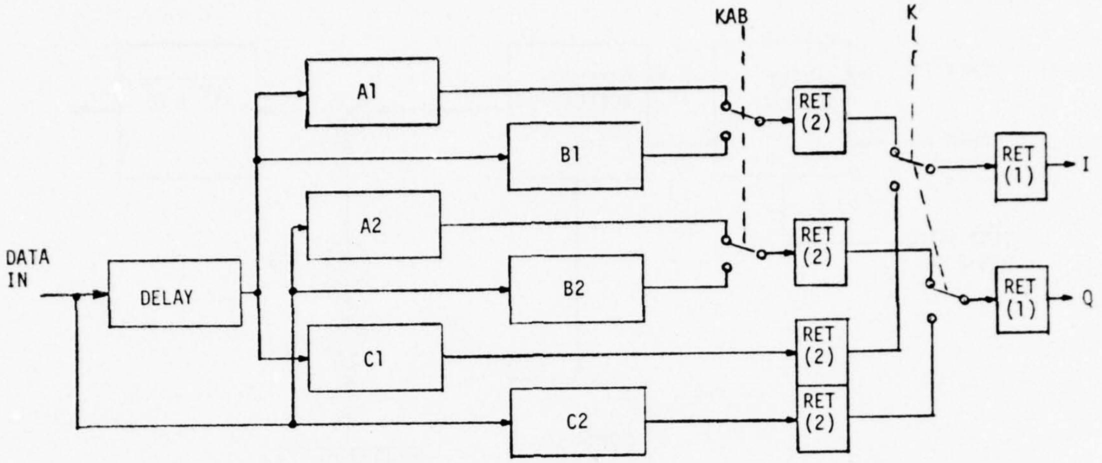


FIGURE 68. INPUT BUFFER BLOCK DIAGRAM

The data input is 7 bits plus a sign bit at a rate of (7/8) 10 MHz = 8.75 MHz. The delay element shown in Figure 68 is an eight bit GUA module which is programmable to give 8, 16, or 32 delays. This splits the aperture for the radix-2 operation.

There are 3 register sets A, B, and C which are programmable in length. The three sets allow for the overlap of input apertures. The length is determined by the aperture length.

The register sets are the two registers on a GUA chip. A1 and B1 share the same chip sets, A2 and B2 share the same chip sets,and C1 and C2 share the same chip sets. This organization allows the output switch of the GUA chip to be utilized for part of the muxing thus eliminating the need for a second control module. However, this requires cascading the GUA's to obtain the required delays. Figure 69 represents the two cascaded registers of the A or B register sets.

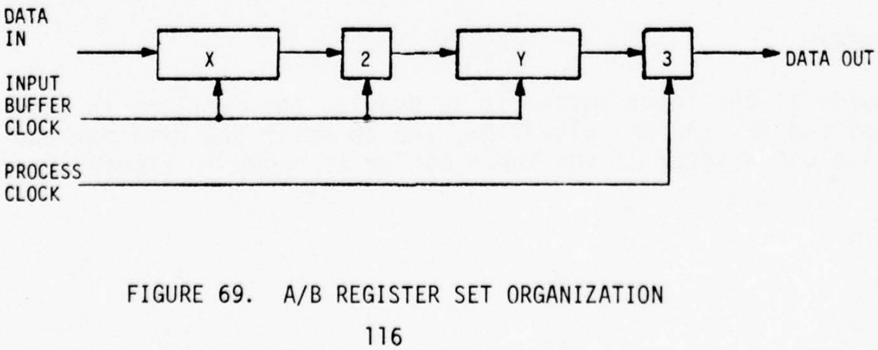


FIGURE 69. A/B REGISTER SET ORGANIZATION

X represents one of the 16 bit registers in a GUA. The block designated 2 represents the 2 delays after the output switch of the GUA. Y represents the 16 bit register of the cascaded GUA. The block marked 3 represents the sum of the 2 delays after the output switch of the GUA associated with Y and the one delay in the control module switch which follows the second GUA. In order to obtain the proper delays for the 3 input modes, the following equations must be satisfied.

1. $X + 2 + Y = 8$
   $X + Y = 6$

2. $X + 2 + Y = 16$
   $X + Y = 14$

3. $X + 2 + Y = 32$
   $X + Y = 30$

Equation #1

Solution	X	Y
1	1	5
2	2	4
3	3	3
4	4	2
5	5	1

Only Solution 2 and Solution 4 represent valid delays for X and Y in the GUA. Either may be chosen.

Equation #2

Choosing only valid values for X, the values for Y are:

Solution	X	Y
1	1	13
2	2	12
3	4	10
4	7	7
5	8	6
6	14	0

Of these, only Solution 4 is a valid delay for Y.

Equation #3

Again, choosing only valid values for X and also keeping in mind X and Y ≤ 16 by definition, then the values of Y are:

Solution	X	Y
1	14	16
2	16	14

Both of these are valid delays for Y.

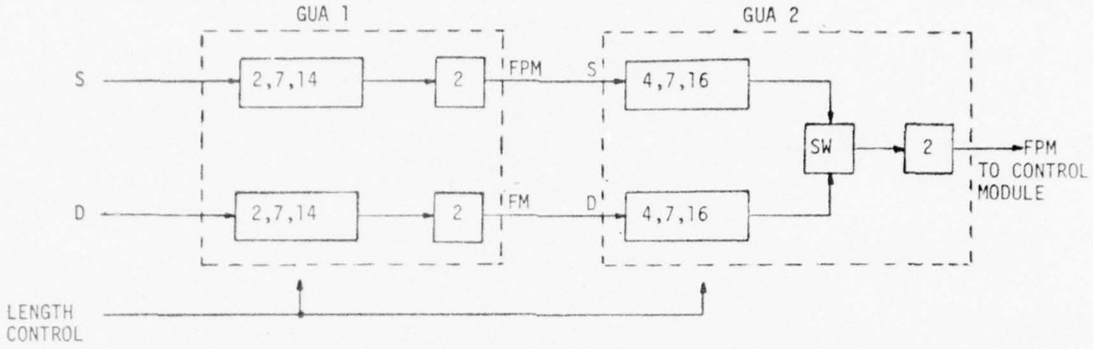


FIGURE 70. A/B REGISTER SET SHOWING VALID DELAYS FOR INPUT BUFFER

The resulting structure is shown in Figure 70. This applies only to the A and B register sets. For the C register set, if the C1 and C2 registers are contained on separate chips, the following lengths must be used:

Let U = Length of the upper register on the chip
L = Length of the lower register on the chip

4. $U + L + 2 = \Delta$ $(\Delta = 8, 16, 32)$

or if the delay switch on the GUA is used

5. $U + L + 4 = \Delta$ $(\Delta = 8, 16, 32)$

U and L must be equal because the U and L registers are on the same chip. Therefore, the above equations become:

6. $L = (\Delta-2)/2 = \Delta/2-1$ $(\Delta = 8, 16, 32)$

and,

7. $L = \Delta/2-2$ $(\Delta = 8, 16, 32)$

Δ	L By Equation 6	L By Equation 7
8	3	2
16	7	6
32	15	14

Since 3, 6, and 15 are not valid GUA delays, then Equation 7 must be used for Δ=8 and Δ=32 and Equation 6 for Δ=16. This means that the delay switch on the GUA must be used.

If the registers are cascaded as in the A and B case, then the delay equation is:

8. $X + Y + 4 = \Delta$ $(\Delta = 8,16,32)$
$X + Y = \Delta - 4 = [4,12,28]$
$[X,Y] = (2,2), (4,8), (14,14)$

The cascaded method eliminates the need for decoding the input stage mode code to produce the DS bit. However, since the modules are cascaded, there is additional inter-module wiring that would not be encountered in the previous case. The decoding of the DS bit can be eliminated by choosing the proper code. Since there are only 3 input cases, a valid 2 bit code is:

Case	Delay	CODE	
		CB	CA
1	8	0	0
2	16	0	1
3	32	1	0

Since a logical 0 at the DS input causes the 4 bit delay in the GUA and a logical 1 causes a 2 bit delay, then choosing the least significant bit of the mode code for the DS bit of the C register set eliminates the need for further decoding.

7.2.1 Data Overlapping

Since the GUA is a shift register and thus continuously loading, the overlap function must be done by selectively looking at the output of the registers and changing clocks. Figure 71 illustrates how data is processed through the input buffer. Consider register set A in Figure 72 in the following discussion.

The case in Figure 71 is for input case 3 where the aperture length is 64 and Δ=32. Thus the length of A1, A2 and DELAY is 32. Data comes into the buffer at data rate 8.75 MHz. If some data word is designated as the start of a window and t_0 is designated as the time at which that word is present at the input of the buffer, then at t_{64} (64 clock pulses later) A1 will contain the first 32 points of the aperture and A2 the second 32 points. At this time, the A set is considered loaded and ready to shift its contents out at the processing rate of 10 MHz. Since 32 clock pulses at the process rate exactly equals 28 clock pulses at the sample rate, then the output process takes 28 sample pulses. However, if the B register set were to start its loading at the time the A register set started processing, it would not be filled with the proper data when A is finished. Now A is finished at $t_{64} + 28 = t_{92}$. It takes 32 pulses to fill B so B must start its fill at $t_{92} - 32 = t_{60}$. The registers do not start filling or stop filling at a particular time, but are continuously filling even when they are dumping. If a word were designated as word 1 and for the shift register lengths stated for case 3, then word 1 would be the first word out of register A1 after t_{64}. Likewise, since A2 has shifted the first 32 words out by t_{64} the word 33 will be the first out of A2 after t_{64}. Likewise for the B register set, word 29 is the first out of B1 and word 61 is the first out of B2. The same goes for register set C. By the time C1 and C2 are ready to dump, they contain the word sets [57-88] and

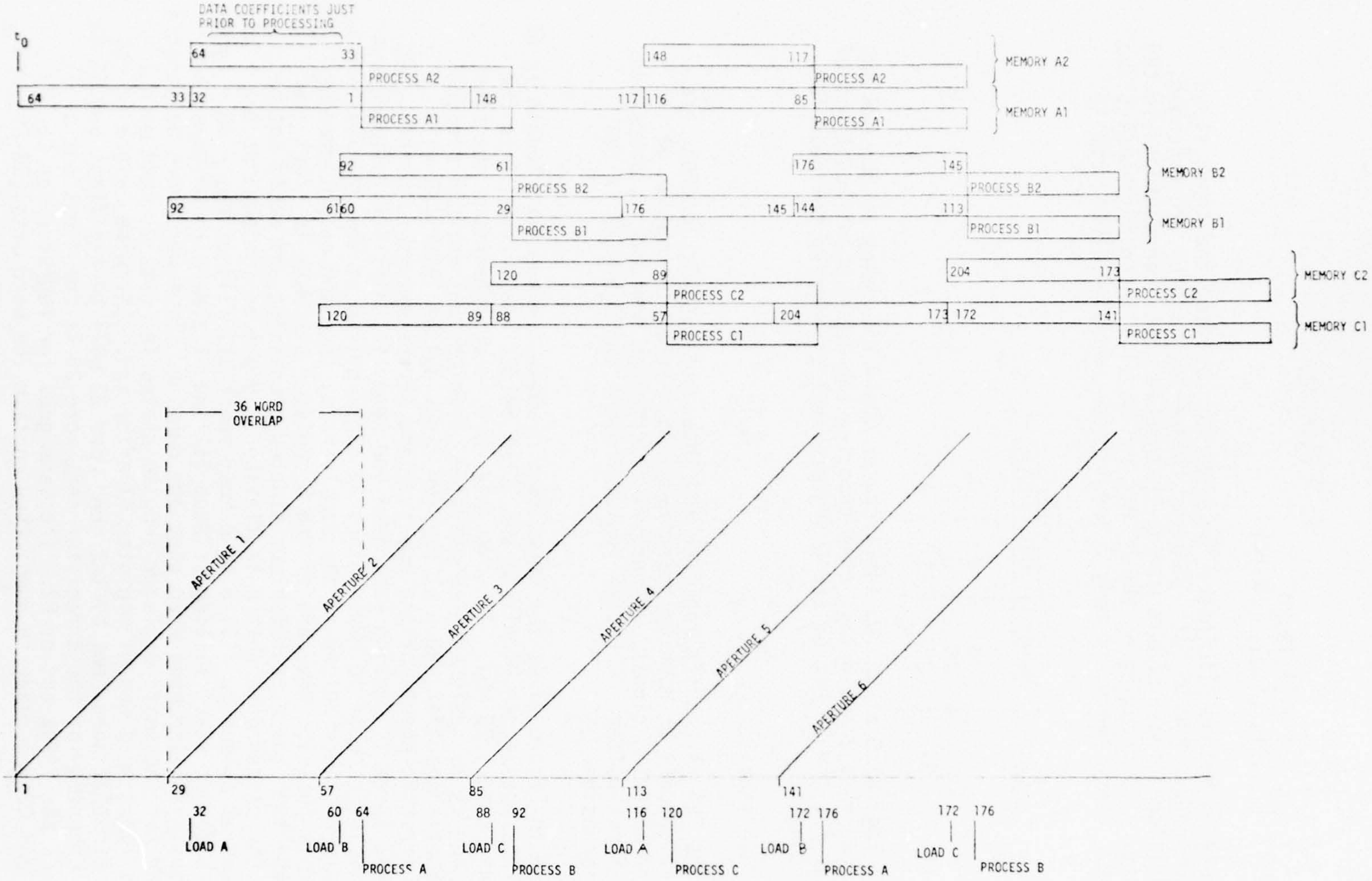


FIGURE 71. DATA OVERLAPPING IN INPUT BUFFER

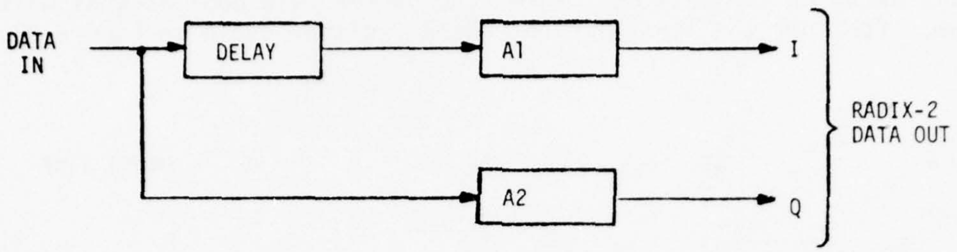


FIGURE 72. REGISTER SET A

[89-120] respectively. This creates a 36 word overlap between apertures in A, B, and C.

In a free running system, the definition of t_0 is unimportant and the input side of the buffer needs no control. The only constraints are on the processing or output of the buffer. These constraints are as follows:

1. The buffer clock must switch at a time when the process and sample clocks are in phase.

2. The length of the buffer must be one half the aperture length.

There are two cases when the system is not free running.

1. When data is being input to the system from the computer in which case the input waveform must be aligned with respect to the starting point of an aperture.

2. When the system is in the waveform generation mode in which case an impulse is to be inserted in the starting point of the first aperture to be processed and zero for all other apertures.

In the first case, the first word of the input waveform must be the first word of the first aperture into the FFT. Referring to Figure 71, this means that the first word of the input waveform must occur at t_0 or t_{28} or t_{56}, etc.

Now a counter must be used on the process side to count the points in an aperture. Constraint number 1 means that the counter must sync or go to 0 when the clocks are "in phase". Also, when the counter goes to 0, this indicates to the control system to select another of the 3 buffer registers. So when the counter syncs, the first word of an aperture is dumped from the buffer. The problem in case 1 is to get the start of the waveform into the first location of one of the 3 registers when the counter syncs. The difficulty lies in the fact that the data enters the buffer at the sample rate (8.75 MHz) and the counter operates at the process rate (10 MHz) and the point in time at which the data must enter the buffer is a non-integral unit of process time. Consider the first part of the A register cycle in Figure 73.

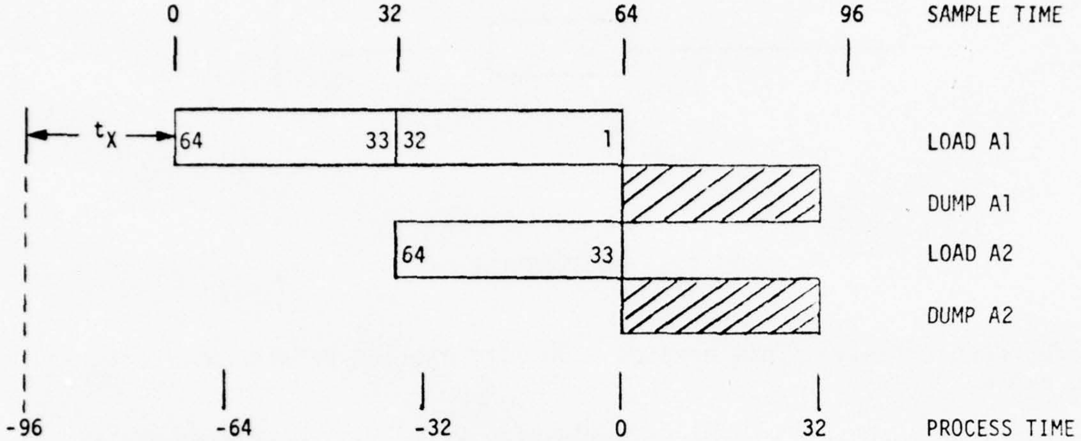


FIGURE 73. REGISTER A CYCLE

For the 64 point aperture waveform, the start of the waveform into the input buffer must be at t_0 for the proper orientation within the aperture. The only points which can be defined as truly common to the two clock systems are at the sync points of the counter. So the only common point before the start of the waveform is at t_{-96} on the process time scale. The variable t_X is the number of pulses after t_{-96} at which the data must be input into the input buffer. More importantly, t_X is in the sample time domain.

t_X represents 5/8 of the length of one half an aperture. Thus, for aperture lengths:

64	$t_X = 1/2\ (5/8)\ (64) = 20$
32	$t_X = 1/2\ (5/8)\ (32) = 10$
16	$t_X = 1/2\ (5/8)\ (16) = 5$

The second non-free-running case is waveform generation in the transmit mode. This is not strictly non-free-running since there is no input other than zero for all time. However, for one pulse at the start of the waveform generation cycle (the first point of the first aperture), an impulse must be inserted in the zero data stream. This can be done by complementing the output of the shift register using the complement control (Figure 74). Note that this is to be a real impulse, not complex. Thus, only the FPM output is to be complemented. Since only the first point of the first aperture is to be complemented, the complement signal must be coordinated with the GO signal from the computer interface.

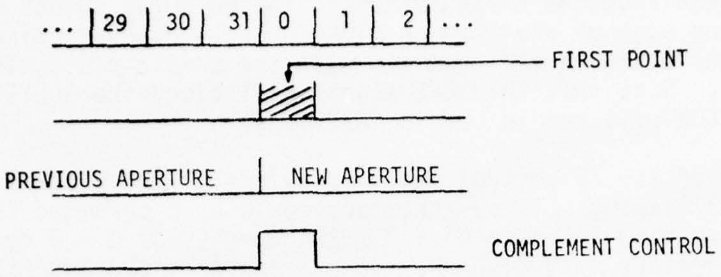


FIGURE 74. COMPLEMENTING THE FIRST POINT IN AN APERTURE

If the GO pulse from the computer is less than 100 nanoseconds, then a simple R-S flip flop can be used to enable the complementer. Figure 75 shows such a circuit.

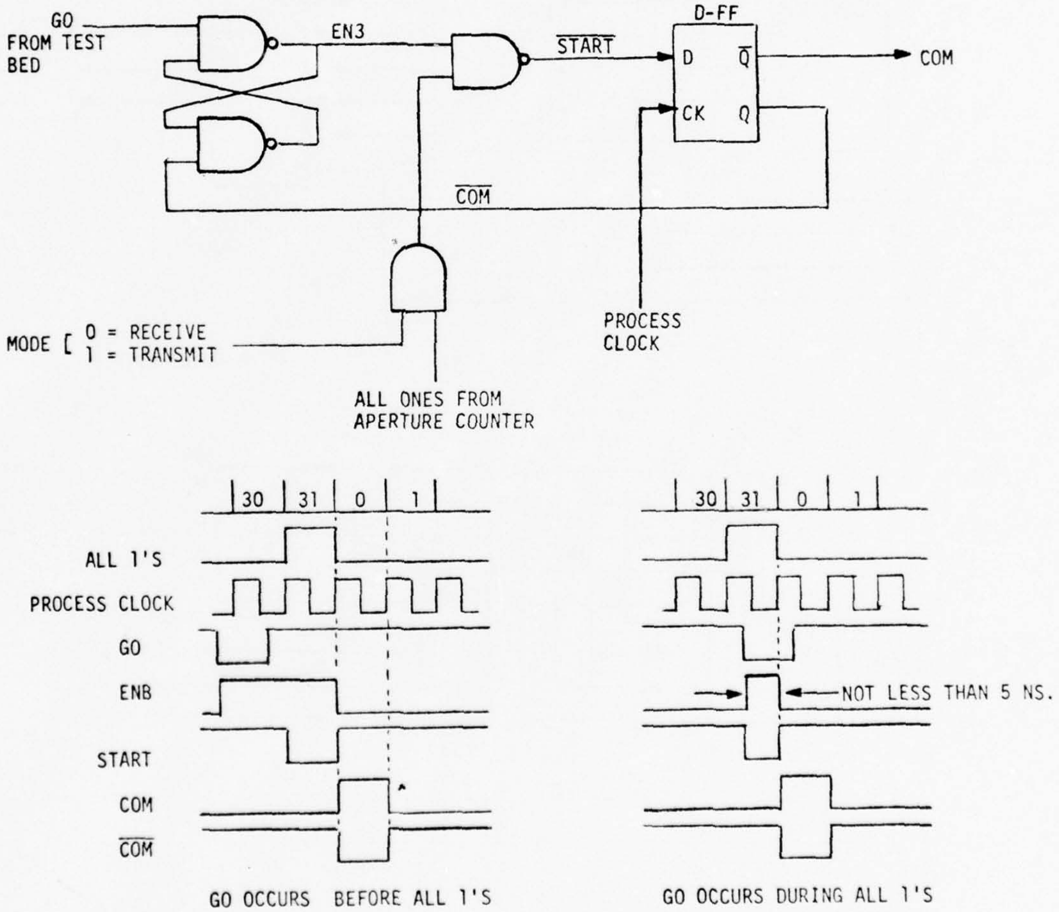


FIGURE 75. IMPULSE GENERATION FOR TRANSMIT MODE

As can be seen from the timing diagram, the GO pulse cannot occur within 5 ns of the rising edge of the inphase clock pulse. However, since the GO pulse is to be reclocked in the test bed with the sample clock, this condition will never occur. Note that the MODE signal will block the ALL 1'S in the receive mode so COM will remain LOW in that mode.

The balance of the IB control system consists of the clock switch and the register multiplexing. These functions could be programmed into a PROM. However, the addressing is based on a 3 x 32, 3 x 16, or 3 x 8 cycle because there are 3 IB register sets A, B, C. This means that the address generating counter must be reset at 96, 48, or 24. The PROM contains the bits required for switching the clocks. Figure 76 shows the PROM organization for the clock switch and the control circuit.

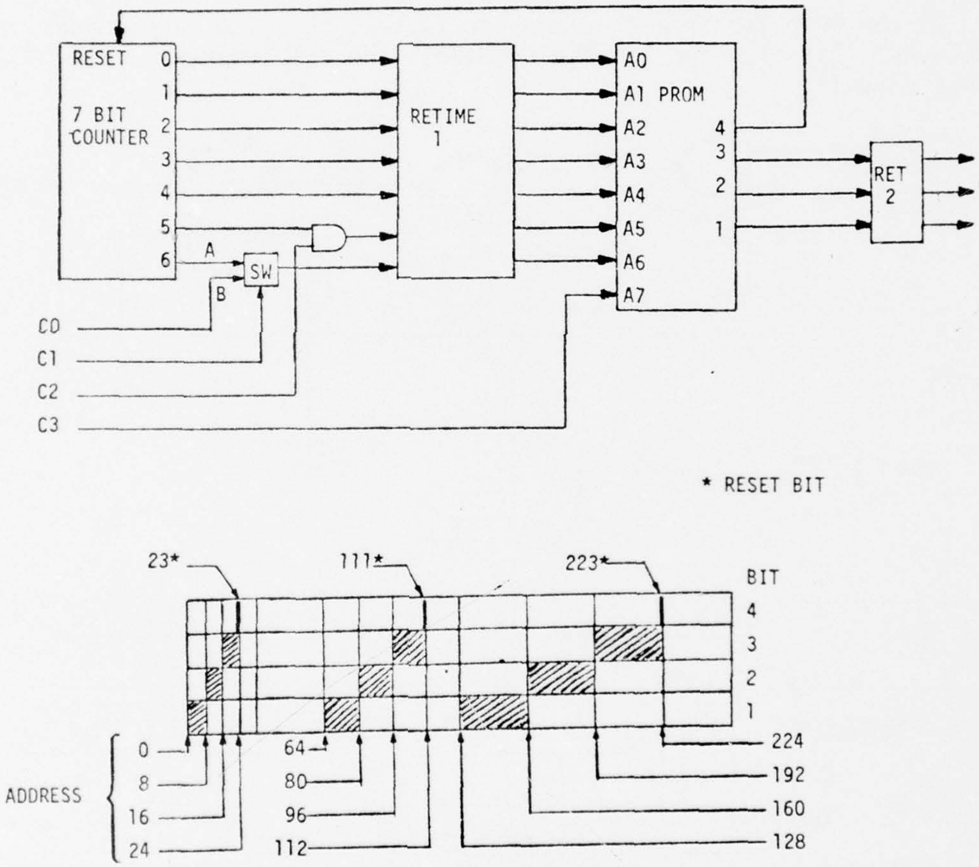


FIGURE 76. PROM ADDRESS GENERATOR AND PROM ORGANIZATION

Figure 77 shows the phase 1 and phase 2 clock switch for the A register of the input buffer. When ECpA, which is the first bit of the PROM, is in the HIGH state, then the output clock is the process clock and when ECpA is LOW, the output is the sample clock. It is this switching between process and sample clock coupled with the muxing control which causes the data overlap.

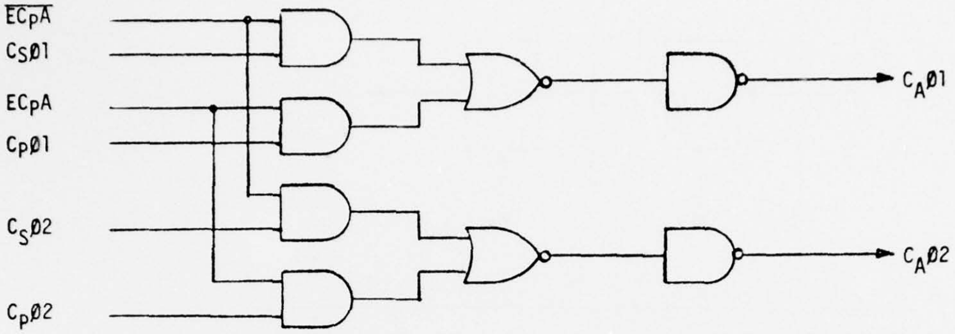


FIGURE 77. CLOCK SWITCHING FOR A REGISTER SET. TYPICAL OF B AND C.

7.2.2 Multiplexing the Input Buffer

Referring back to Figure 68, it can be seen that there are two sets of switches in the input buffer. The first set is the output switch of the A,B register set. The second is the select switch of the control module. Note that the clock switching takes place for the registers up to the output switch of the GUA. The 2 delays after the switch are continuously clocked at the process rate and t_0 for the system timing is at this point. However, the IB consists of the GUA's and the control module so data comes out of the IB at t_2.

Now at the end of an input cycle say for the A register, the next clock to come along will be the first output clock.

Care must be taken in defining the inphase point of the clock. Due to the dual phase requirements of the GUA, there are 2 points which can be defined as the inphase point. Data is loaded into stage 1 of a GUA register cell when Ø1 goes LOW and data appears at the output when Ø2 goes LOW.

Consider the clock pulse string in Figure 78. Data appears at B when Ø1 goes LOW and is held until Ø1 goes LOW again. However, while Ø1 is LOW A=B so data must not change until after Ø1 goes HIGH. Data then appears at C when Ø2 goes LOW. Now the LOW to HIGH transition of the Ø1 clock appears to be the logical choice for the reference edge. If this is so, then the propagation delay of the GUA cell can be considered to be:

$$t_p = t_0 = \text{the overlaps}$$

The advantages of this choice are:

Another approach would be to partially perform m(t) in the memory so that

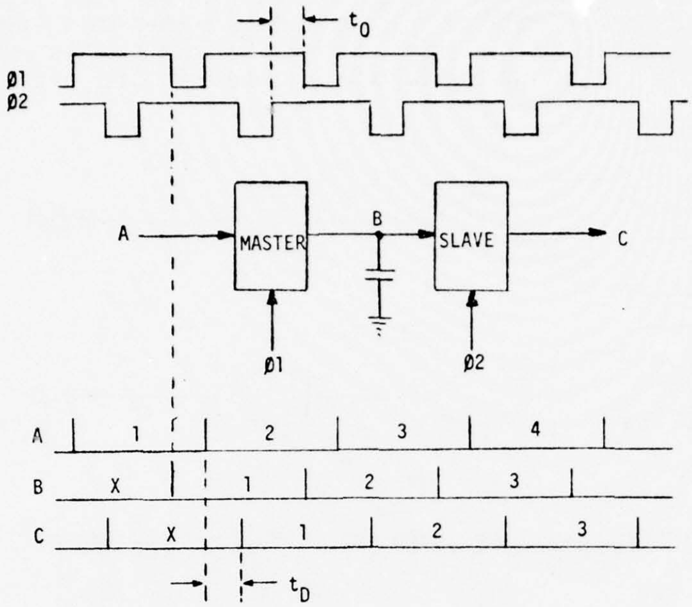


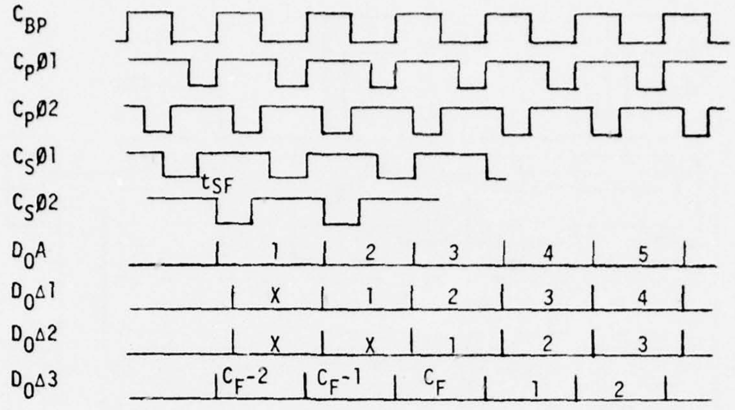
FIGURE 78. CLOCK REFERENCE EDGE

1. Retimer registers output on the rising edge of the clock.
2. The rising edge of the clock is the natural reference for TTL devices.
3. The phase locked loop oscillator locks on a rising edge.

So now the timing diagram for the input buffer can be defined with respect to this point (Figure 79).

Now at t_{SP}, which is the last Ø2 of the A register load cycle, the first point of an aperture is at the output of the last register cell of A. On the next clock pulse, data will start out of register A at the process rate. Since the last 2 retimers are GUA cells, then data 1 must be through the switch and set up in the master side of the cell before C_pØ1 goes high at t_0. This means that the output switch of the A-B GUA must switch to the A position after t_{SF} and before t_0. Since the control is TTL, the most logical place is t - 1/2. However, it is desirable to have the switch change at the same time as the clocks change. Since we do not wish to add any spikes, the clock must be switched when both C_SØ1 and C_pØ1 are low. Due to the fact that the low time of C_SØ1 is longer than C_pØ1, we can just say that the switch must switch when C_pØ1 goes low before t_0. Thus, no effect will be made on the clocks since the last cell of A will be loading with data point 2 and will continue to load, and the Δ1 register will be in the master load condition long enough to allow data 1 to propagate through the switch and set up in the Δ1 interstage capacitive memory.

In order to get A(t), let us generalize the function p(t,ℓ).



C_{BP} = BASE PROCESSING CLOCK (TTL)

$C_P\emptyset 1$ = PROCESS CLOCK Ø1

$C_P\emptyset 2$ = PROCESS CLOCK Ø2

$C_S\emptyset 1$ = SAMPLE CLOCK Ø1

$C_S\emptyset 2$ = SAMPLE CLOCK Ø2

D_0A = DATA OUTPUT OF REGISTER SET A

$D_0\Delta 1$, $D_0\Delta 2$, $D_0\Delta 3$ = DATA OUTPUT OF REGISTERS Δ1, Δ2, Δ3

FIGURE 79. INPUT BUFFER TIMING

Now referring back to Figure 76, it can be seen that if the information in the PROM is rotated back one address location, i.e., D(192) becomes D(191), D(223) becomes D(222), D(0) becomes D(223), etc., and if the retimer number 2 is clocked with the inverse of CpØ1, then the switch will switch at the proper time and the clock will switch at the proper time.

Refer again to the timing diagram in Figure 79. At t_1 data 1 appears at the output of the GUA or register Δ2 and the output of register Δ3 is C_F (the last point in the previous aperture). Between this clock pulse and the next, the switch on the control module must switch. Since Δ3 is a retimer register and requires a monophase clock (Ø1) and since the master loads when the clock is low, then the data cannot change 15 ns before the end of the Ø1 low period. Since the Δ3 register output is C_F at t, which coincides with the leading edge of $C_{BP}(t)$, then $C_{BP}(t)$ is used to clock the switch control pulse out.

7.2.3 Summary of the Input Buffer and Control System

A detail of the input buffer and its control inputs is shown in Figure 80. The input buffer consists of 6 sets of GUA circuits with the organization shown

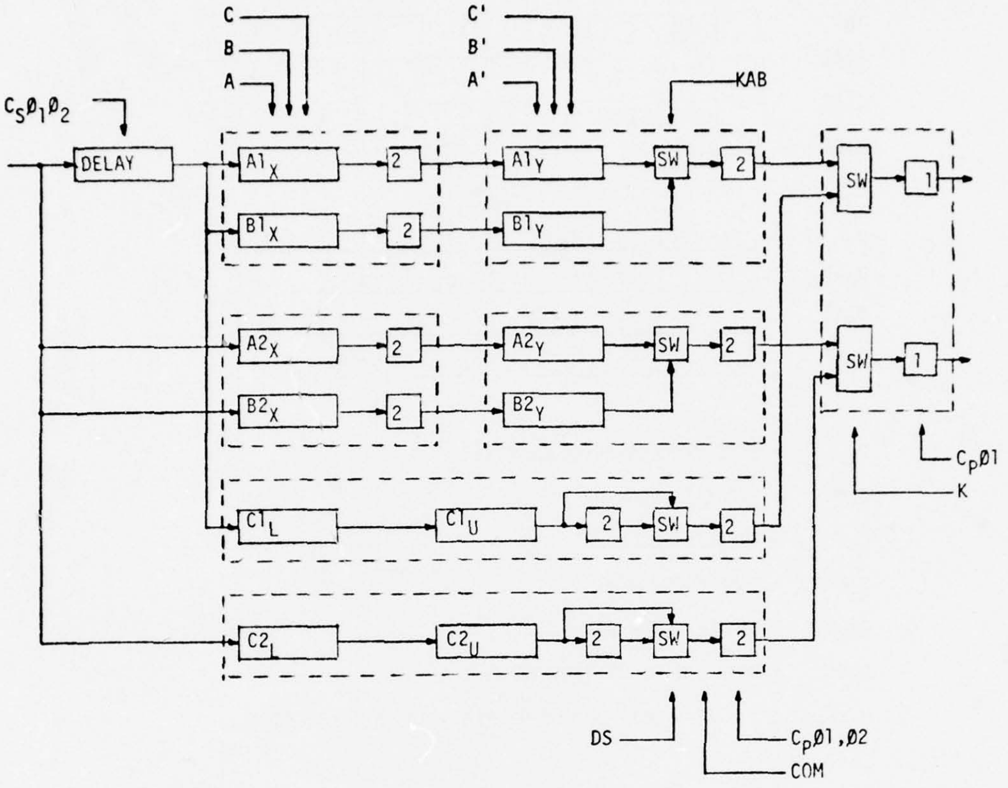


FIGURE 80. INPUT BUFFER WITH CONTROL SIGNALS

in the figure. Each set consists of 2 modules of 8 bits each; one for the real and one for the imaginary input. The control module set at the input buffer output consists of 2 modules and the delay set at the input consists of 2 modules. In addition to these modules, there are 2 complex multipliers for ramping, 7 clock drivers, 3 universal TTL modules containing the control circuits, 2 universal TTL modules for the ramping PROM's and 2 level translators for the ramping data. The module totals are:

14	-	GUA Modules
2	-	Control Modules
2	-	Complex Multipliers
2	-	Level Translators
7	-	Universal TTL Clock Drivers
5	-	Universal TTL Control Functions
32	-	Total Modules

A block diagram of the control system with all the control signals is shown in Figure 81. The signals C_A and C_B are inputs from the test bed REG8. The GO signal is also from the test bed and is a response to the START instruction given by the computer.

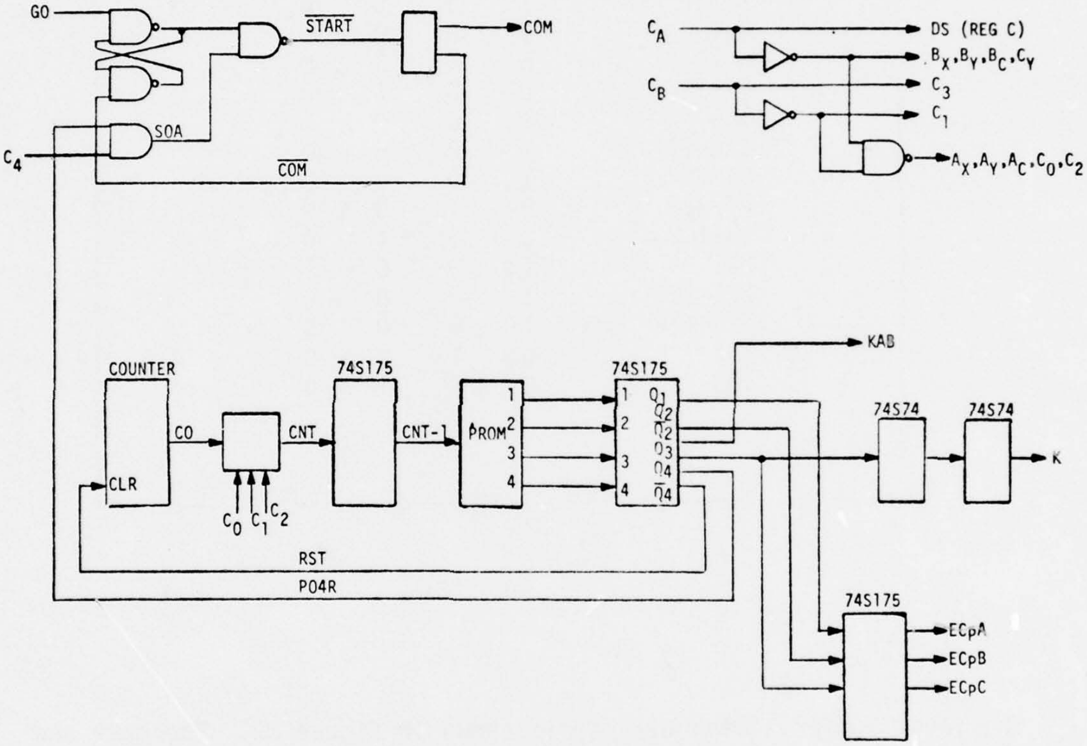


FIGURE 81. INPUT BUFFER CONTROL SYSTEM

The control PROM contents are shown in Table 31. These bits are the ECpA, ECpB, and ECpC signals for switching the clocks. The multiplex controls KAB and K are derived from these signals and have the relationship:

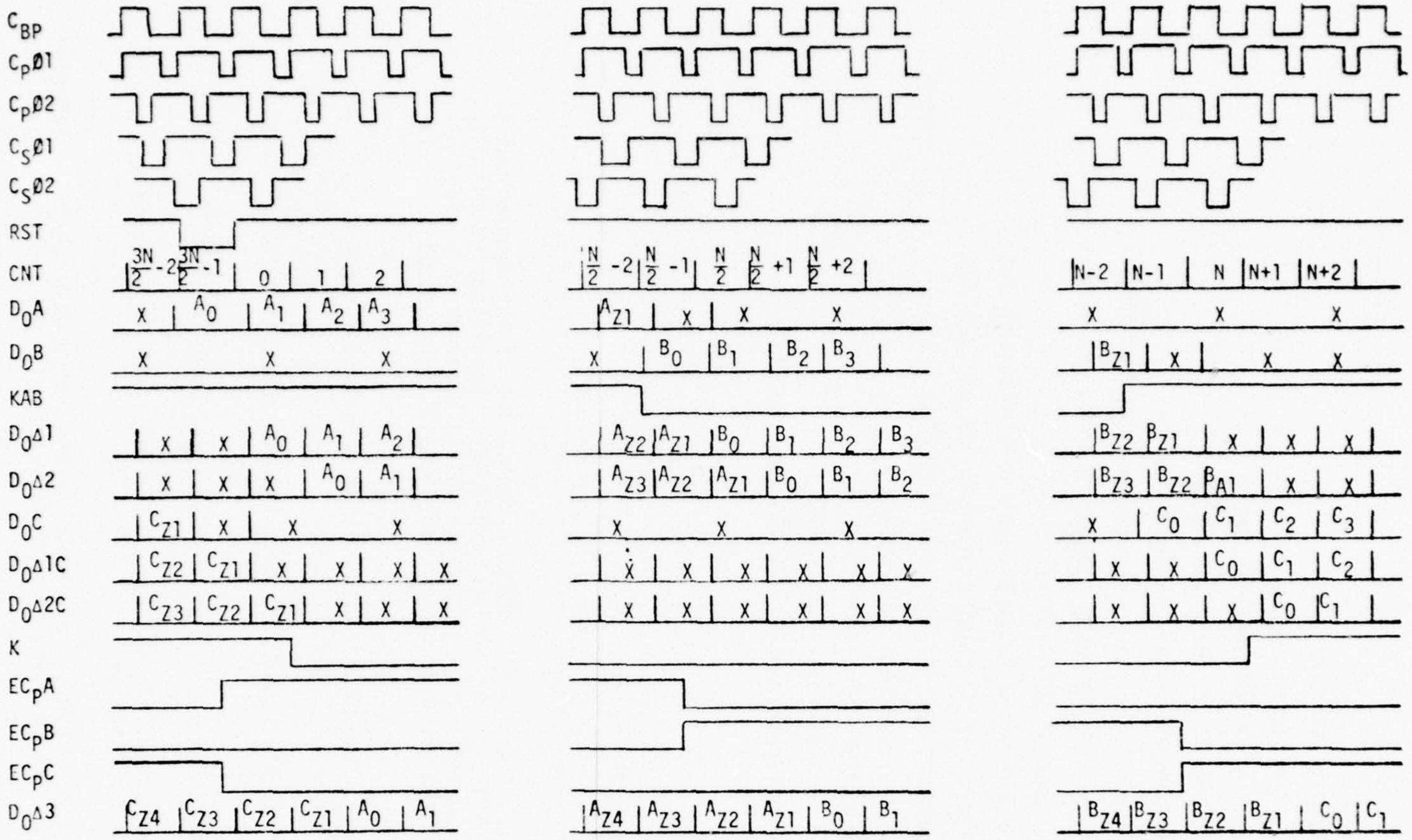
$$\text{KAB} = \overline{\text{ECpB}}$$
$$\text{K} = \text{ECpC}$$

TABLE 31
IB CONTROL PROM CONTENT

PROM				
	CONTENT			
ADDRESS	1	2	3	4
21	1	0	0	1
22-23	1	0	0	0
0-4	1	0	0	0
5-12	0	1	0	0
13-20	0	0	1	0
109	1	0	0	1
110-111	1	0	0	0
64-76	1	0	0	0
77-92	0	1	0	0
93-108	0	0	1	0
221	1	0	0	1
222-223	1	0	0	0
128-156	1	0	0	0
157-188	0	1	0	0
189-220	0	0	1	0

The input buffer timing diagram is shown in Figure 82. The data and control signals shown there are as follows:

C_{BP} - Base Processing Clock
$C_{P}\emptyset 1$ - Process Clock Phase 1
$C_{P}\emptyset 2$ - Process Clock Phase 2
$C_{S}\emptyset 1$ - Sample Clock Phase 1
$C_{S}\emptyset 2$ - Sample Clock Phase 2


C_{BP}
$C_PØ1$
$C_PØ2$
$C_SØ1$
$C_SØ2$
RST
CNT
D_0A
D_0B
KAB
$D_0\Delta 1$
$D_0\Delta 2$
D_0C
$D_0\Delta 1C$
$D_0\Delta 2C$
K
EC_PA
EC_PB
EC_PC
$D_0\Delta 3$
$\frac{3N}{2}-2$ $\frac{3N}{2}-1$ 0 1 2
$\frac{N}{2}-2$ $\frac{N}{2}-1$ $\frac{N}{2}$ $\frac{N}{2}+1$ $\frac{N}{2}+2$
N-2 N-1 N N+1 N+2
X A_0 A_1 A_2 A_3
A_{Z1} X X X
B_0 B_1 B_2 B_3
B_{Z1}
A_{Z2} A_{Z1} B_0 B_1 B_2 B_3
A_{Z3} A_{Z2} A_{Z1} B_0 B_1 B_2
B_{Z2} B_{Z1}
B_{Z3} B_{Z2} B_{A1}
C_{Z1} C_{Z2} C_{Z3}
C_0 C_1 C_2 C_3
C_{Z4} C_{Z3} C_{Z2} C_{Z1} A_0 A_1
A_{Z4} A_{Z3} A_{Z2} A_{Z1} B_0 B_1
B_{Z4} B_{Z3} B_{Z2} B_{Z1} C_0 C_1

$ZR = \frac{N}{2} - R$ WHERE N IS THE APERTURE LENGTH AND R IS THE TIME INDEX

FIGURE 82. INPUT BUFFER CONTROL SYSTEM TIMING

RST	Control Counter Reset
CNT	Control Counter Output
D_0A	Output of GUA Register Set A
D_0B	Output of GUA Register Set B
KAB	A/B Register Set Multiplex Control
$D_0\Delta 1$	Output of $\Delta 1$ Delay Element on A/B 1 GUA Chip Set
$D_0\Delta 2$	Output of $\Delta 2$ Delay Element on A/B 2 GUA Chip Set
D_0C	Output of GUA Register Set C
$D_0\Delta 1C$	Output of $\Delta 1$ Delay Element on C1 GUA Chip Set
$D_0\Delta 2C$	Output of $\Delta 2$ Delay Element on C2 GUA Chip Set
K	Control Module Multiplex Control
EC_pA	A Register Set Clock Switch
EC_pB	B Register Set Clock Switch
EC_pC	C Register Set Clock Switch
COM	Complement Control for Register Set A/B 1

7.3 REORDER MEMORY CONTROL SYSTEM

7.3.1 Reorder Memory Requirements

The step transform processor requires a reordering of the samples out of the first FFT before they are fed to the second FFT. Specifically, the data samples must be transformed from a column-row matrix sequence to a diagonal across the matrix. There are a number of requirements which complicate the reorder memory design:

1. The input data from the first FFT is in an unnatural (bit reversed) sequence.

2. The data fed to the second FFT must also be in a bit reversed sequence.

3. The system must handle diagonals with slopes of 1 and 1/2.

4. The input and output are two parallel data streams corresponding to the radix-2 FFT processor.

The first three requirements are handled by the addressing scheme implemented in the control system. The last requirement is handled by the architecture of the reorder memory.

7.3.2 Double Multiplex Implementation

In a double multiplex implementation, a total of 8 memories are employed so the read/write cycles can operate at one-half the basic clock speed. A block diagram of the system is shown in Figure 83. The two input channels are fed to alternate registers which hold the data for two clock periods. These registers each feed two memory units which alternate on write and read cycles. Figure 84 shows the basic timing diagram for the system. The read mode precedes the write mode during operation. Therefore, the read addresses are sequenced in natural order directly from the control counter. Each memory unit is started in the read mode at the first memory word. The read addresses for each memory then occur with a period of 400 nsec. The total span of the first read cycle for the four memories is 500 nsec making it necessary to provide an interim store of the read addresses for the "B" memories.

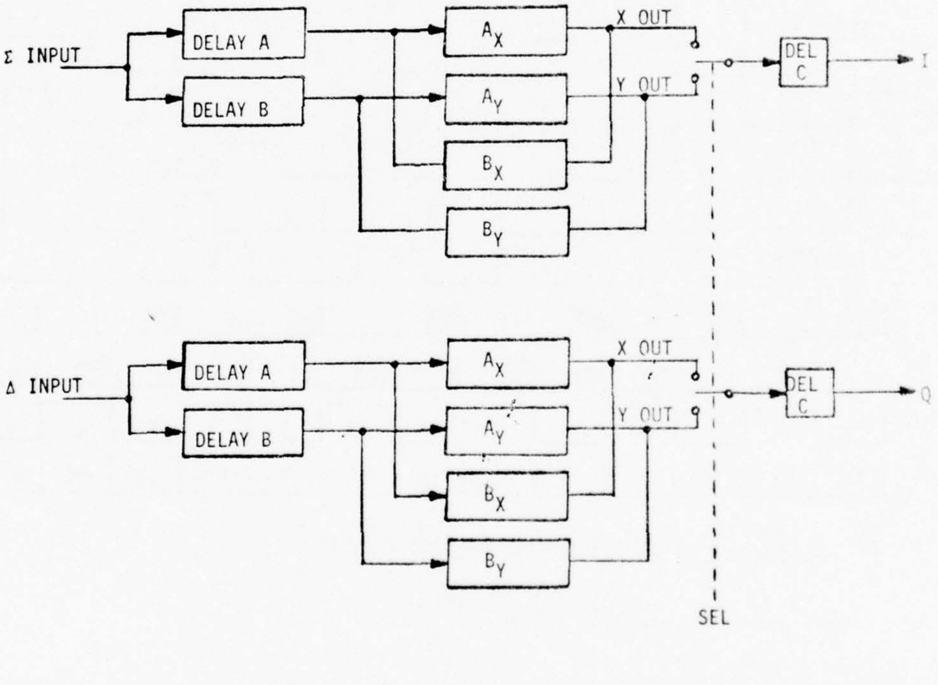


FIGURE 83. REORDER MEMORY ORGANIZATION

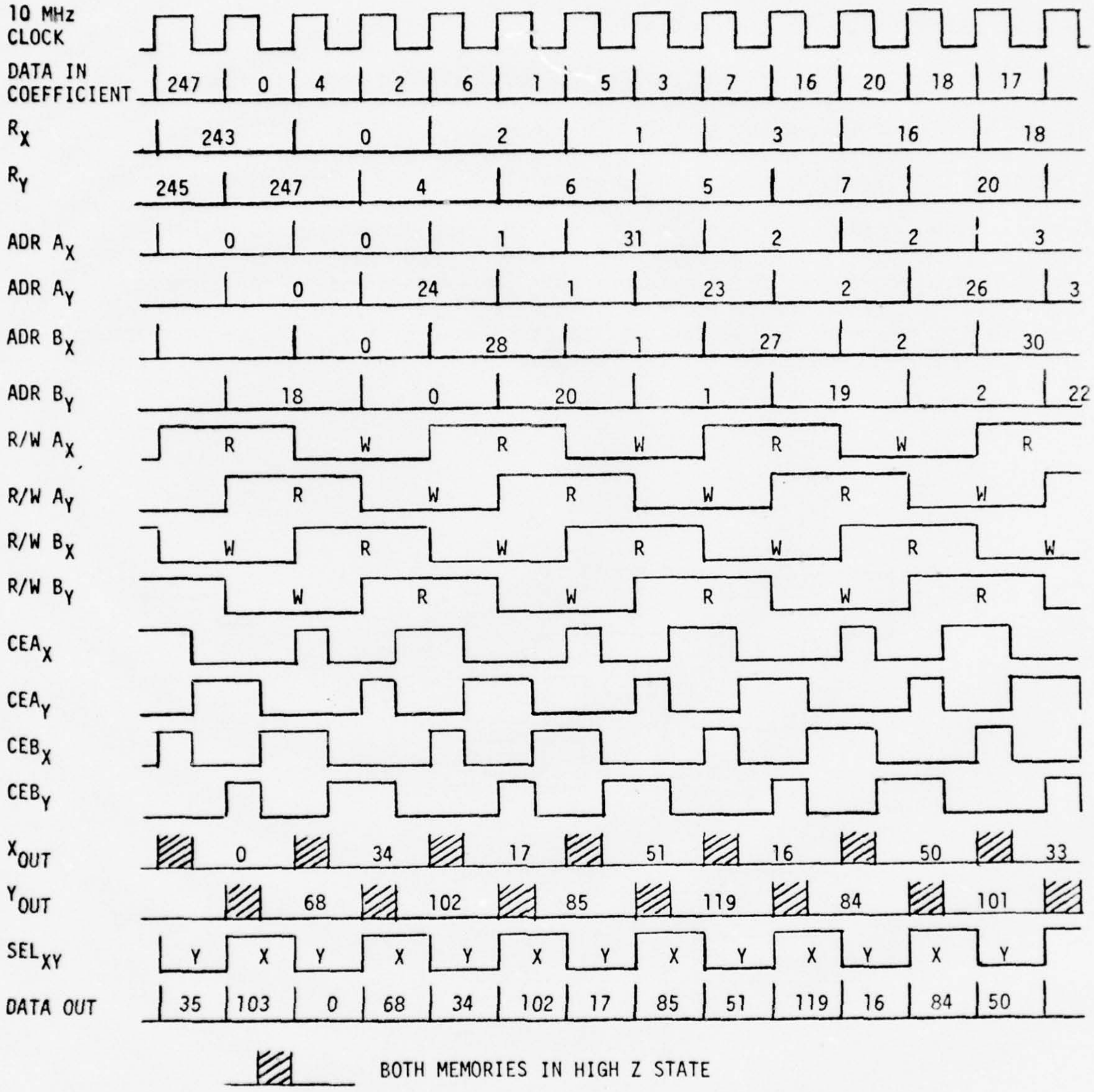


FIGURE 84. REORDER MEMORY TIMING DIAGRAM FOR CASE 1
(N=16, b=1) Σ CHANNEL

The two memory groups numbered with the prefix 1 and 2 handle the Σ and Δ FFT outputs respectively. For the double multiplex operation, the memory capacity is given by:

$$1/b\ (N(N/2)) = N^2/2b \text{ for the } \Sigma \text{ channel}$$

and

$$1/2\ (N^2/2b) = N^2/4b \text{ for the } \Delta \text{ channel}$$

where N is the number of samples in an input aperture and b is the slope of the diagonal.

Since there are 4 memories in each of the Σ and Δ channels, then the storage capacity of each memory need only be one quarter of the total. Table 32 summarizes the storage requirements for each of the cases encountered in the PWP.

TABLE 32
REORDER MEMORY STORAGE

CASE	SAMPLES IN INPUT APERTURE N	SAMPLES IN OUTPUT APERTURE	SLOPE b	TOTAL STORAGE	STORAGE/ MEMORY	
					Σ	Δ
1	16	16	1	128	32	16
2	16	32	1/2	256	64	32
3	32	32	1	512	128	64
4	32	64	1/2	1024	256	128
5	64	64	1	2048	512	256

There are several signals which must be generated by the control system. The LOAD X and LOAD Y, CS and R/W signals are generated by decoding a counter and combining with the clock. The addresses, however, are computed. The read address is a straight sequential address and the write address is computed from a stored base address.

7.3.3 Address Generation

There are 2 addresses being generated for the memories.

1. Read Address
2. Write Address

The read address is in sequential order and spans 5 clock cycles. See the timing diagram (Figure 84). The read address is common to all memories. The write address is unique to each memory, must be generated at the system clock rate and must be held at each memory input for 2 clock cycles. The write addresses are generated using a base address table contained in a PROM and adding successive multiples of a constant K for successive apertures.

Case	N	b	K	ℓ
1	16	1	2	8
2	16	1/2	2	8
3	32	1	4	16
4	32	1/2	4	16
5	64	1	8	32

ℓ = Number of points in the base address sequence

Let I(t) be defined as the greatest integer function of t.

e.g., $5 = I(5.25)$

Also let,

1. $m(t,\ell) = \frac{\ell}{4} I(t/\ell) \qquad t = 0,1,2,\ldots,N^2/2$

and,

2. $p(t,\ell) = t - \ell\, I(t/\ell) \qquad t = 0,1,2,\ldots,N^2/2$

Then the calculated address is given by:

3. $A^1(t,\ell) = B[p(t,\ell)] + m(t,\ell)$

where B[n] is the base address whose address is n. The base address sequence is given in Section 9.3, Reorder Memory Software.

e.g., $N=16 \qquad \ell=8 \qquad t=33 \qquad b=1$

$$m(33) = \frac{8}{4} I\,[(33)/8] = 2(4) = 8$$

$$p(33) = 33 - 8\,I(33/8) = 33 - 8(4) = 1$$

$$A^1(33,8) = B[1] + 8$$

$$= 25 + 8 = 33$$

Now the address cannot exceed the maximum length of the memory for that case so,

4. $A_\Sigma(t) \equiv A'_\Sigma(t,\ell) \bmod (N^2/8b)$

5. $A_\Delta(t) \equiv A'_\Delta(t,\ell) \bmod 1/2\ (N^2/8b)$

so for the above example, b = 1,

$A_\Sigma(t) \equiv 33 \quad \bmod (16^2/8)$

$$A_{\Sigma}(t) \equiv 33 \mod (32)$$

$$A_{\Sigma}(t) = 1$$

This answer corresponds to the address given in the case number 1 address table in Section 9.3.

The control system must generate the two components of $A'(t)$ given ℓ

$$m(t) = \frac{\ell}{4} I(t/\ell)$$

and

$$6. \quad B[p(t)] = B(t - \ell\, I(t/\ell))]$$

for $\ell = 8,16,32$
$t = 0,1,2,\ldots,N^2/2$

t is generated in a counter which is referenced to the first point in the first aperture. The maximum count will be $(64)^2/2 = 2048$.

Thus, the counter must be 11 bits. $I(t/\ell)$ can be obtained by scaling the counter output by M bits to the left.

$$C_0\ C_1\ C_2\ \underbrace{C_3\ C_4\ C_5\ C_6\ C_7\ C_8\ C_9\ C_{10}}_{I_0\ I_1\ I_2\ I_3\ I_4\ I_5\ I_6\ I_7}$$

$\ell = 8$

Now ℓ can be represented by,

$$\ell = 2^M$$

$$\therefore M = \log_2 \ell = [3,4,5]$$

referenced to the minimum value of M and dropping the 3 least significant bits,

$$M' = (\log_2 \ell) - 3 = [0,1,2]$$

and instead of scaling C_0-C_{10}, we scale I_0-I_7.

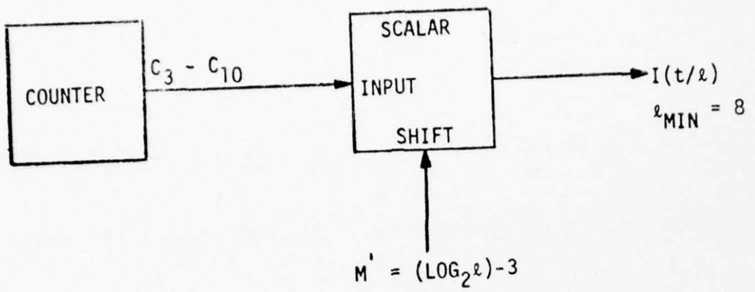


FIGURE 85. SCALER IMPLEMENTATION FOR OBTAINING $I(t/\ell)$ FROM COUNTER OUTPUT

Now $m(t) = \ell/4 \; I(t/\ell)$ so the scaler output must be multiplied by $\ell/4$. This can be accomplished by scaling $I(t/\ell)$ to the right.

$$\frac{\ell}{4} = 2^{M-2}$$

then $M'' = M-2 = [1,2,3]$; Again referencing this to M''_{MIN} we get $M''' = M' =$ [0,1,2] and I_k becomes I'_{k+1} where k = 0,1,2,...,6.

$$I'_0 = 0$$

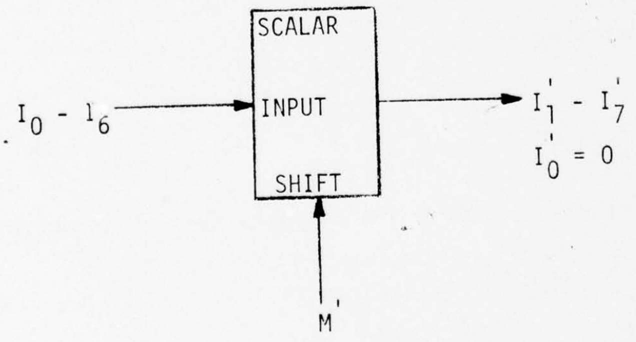


FIGURE 86. GENERATING I' FROM I

This scalar approach is very costly in terms of hardware since scaling in TTL requires quite a few chips.

Another approach would be to partially perform m(t) in the memory so that all that need be added to the base address is a multiple of $\ell_{MAX}/4 = 8$. This is possible because for every four cycles of the base address in cases 1 and 2, the augend is a multiple of 8 and for cases 3 and 4 it is every other cycle.

The number N_C of augmented base address cycles that need be stored then are:

7. $N_C = (\ell_{MAX}/4)/(\ell/4)$

$$= \ell_{MAX}/\ell = 32/\ell$$

Thus the new $A'(t)$ becomes

8. $A''(t) = B'[p_{MAX}(t)] + m_{MAX}(t)$

9. $m_{MAX}(t) = \ell_{MAX}/4 \ I \ (t/\ell_{MAX})$

$$= 8 \ I \ (t/32)$$

10. $P_{MAX}(t) = t - \ell_{MAX} \ I \ (t/\ell_{MAX})$

$$= t - 32 \ I \ (t/32)$$

and

11. $B'[P_{MAX}(t)] = B[p(t)] + m(p_{MAX}(t))$

$$= B[p(t)] + \ell/4 \ I \ [(t-32I(t/32))/\ell]$$

It can be shown that $m(p_{MAX}(t))$ is 0 for case 5, [0,4] for case 3 and 4, and [0,2,4,6] for case 1.

12. $\cdot\cdot A''(t) \equiv A'(t)$

The address generator now looks like the circuit in Figure 87.

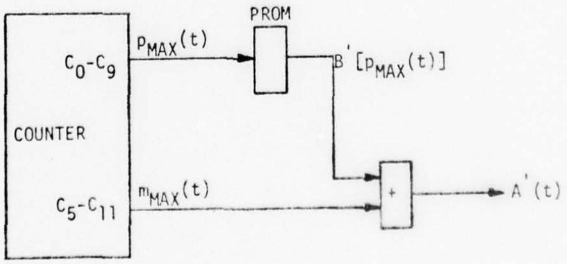


FIGURE 87. BASIC WRITE ADDRESS GENERATOR

In order to get A(t), let us generalize the function $p(t,\ell)$.

13. $q(n,m) = n-m\ I\ (n/m)$

now let $n = A'_{\Sigma}(t)$

14. $m = \frac{N^2}{8b}$

and we get,

15. $A_{\Sigma}(t) = q(n,m) = A'_{\Sigma}(t) - \frac{N^2}{8b}\ I\ [\frac{8b\ A'_{\Sigma}(t)}{N^2}]$

The nice thing about this is that $A_{\Sigma}(t)$ can be obtained by blanking all those bits of $A'_{\Sigma}(t)$ greater than,

16. $M_{\Sigma} = \log_2 \frac{N^2}{8b}$

The complete write address generator for the Σ channel is shown in Figure 88. The gating required for blanking those bits greater than M_{Σ} is shown in Figure 89.

Looking at the tables of Σ and Δ base addresses, the following relationship is evident for each case:

Case	N	b	A_{Σ} Minus A_{Δ}
1	16	1	16
2	16	1/2	32
3	32	1	64
4	32	1/2	128
5	64	1	256

Since the maximum Δ memory length is the same as A_{Σ}-A_{Δ}, then A_{Δ} can be formed from A_{Σ} by dropping the most significant bit of the A_{Σ} address.

The Δ write address gating is shown in Figure 89. The complete control system block diagram with all read/write chip select and address controls is shown in Figure 90.

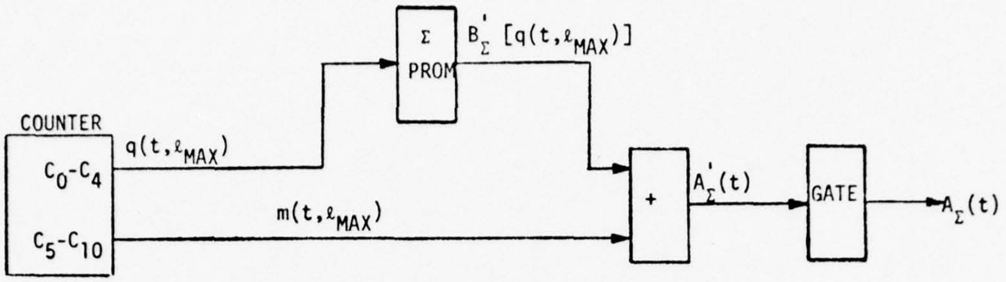


FIGURE 88. BLOCK DIAGRAM OF Σ WRITE ADDRESS GENERATOR

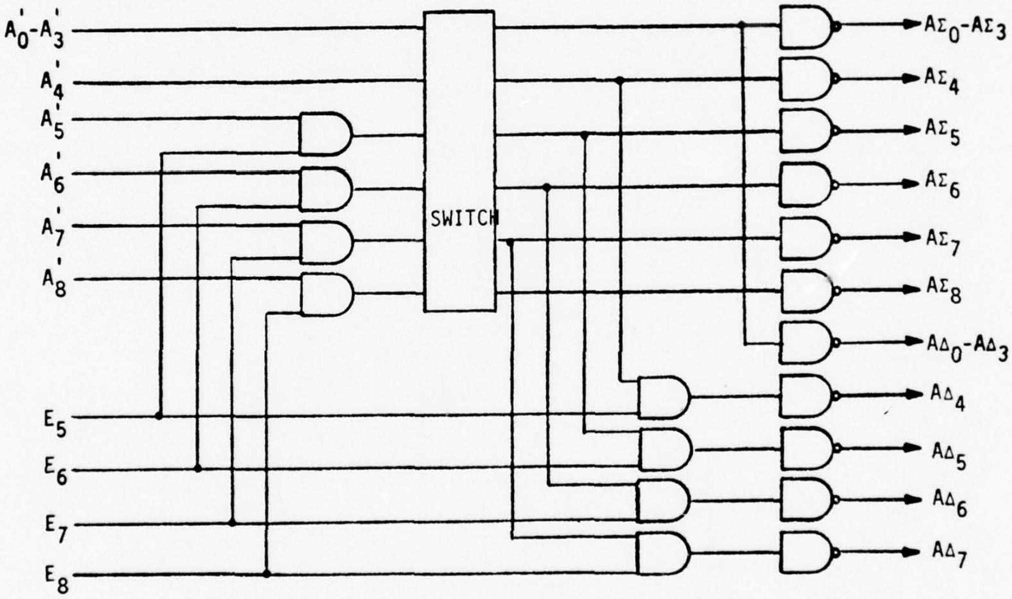


FIGURE 89. GATING TO OBTAIN Δ RAM ADDRESS FROM Σ RAM ADDRESS

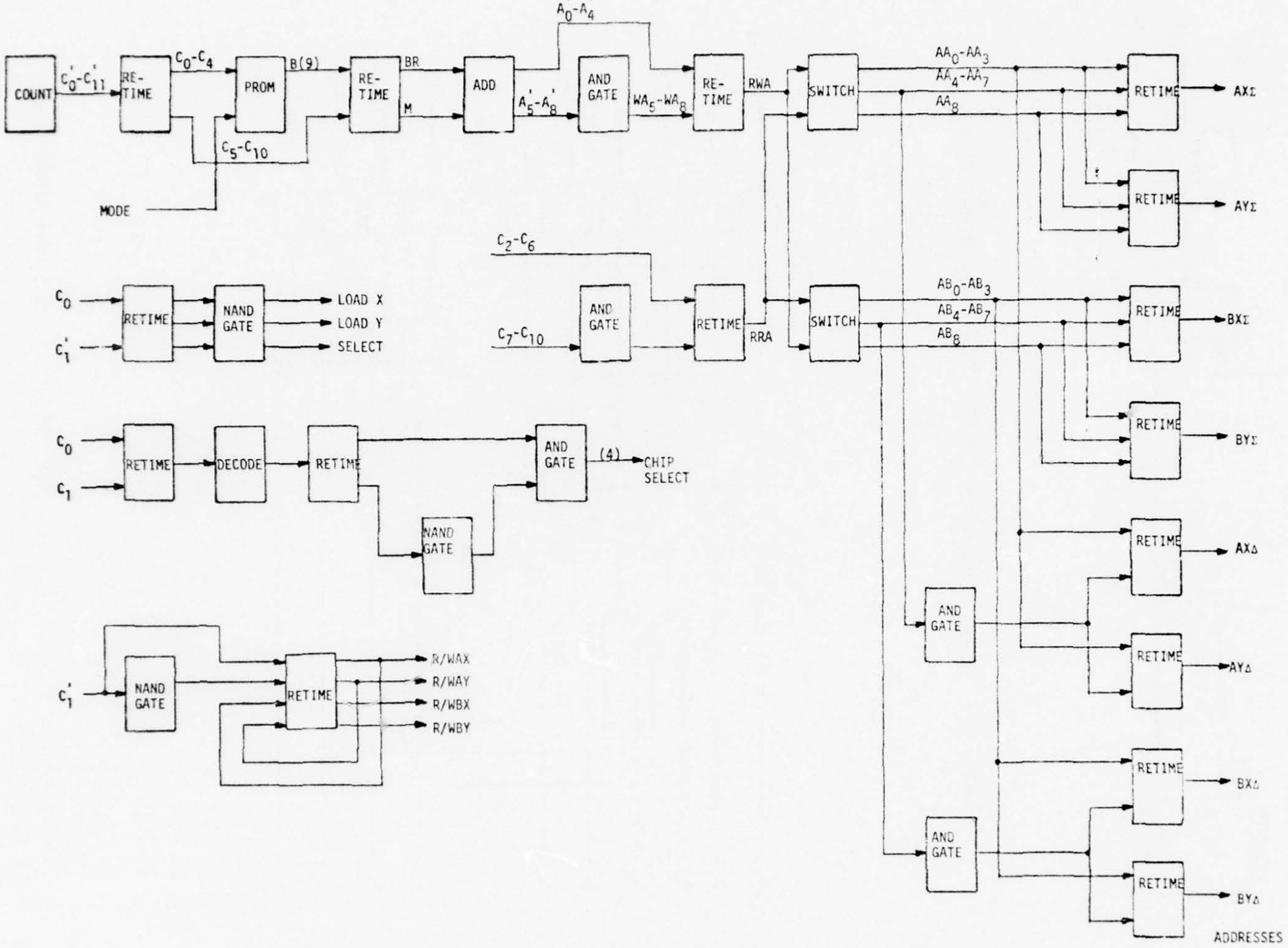


FIGURE 90. REORDER MEMORY CONTROL SYSTEM

7.4 VARIABLE LENGTH PIPELINE FFT'S

The forward and inverse FFT's are identical except for the switching required to make them programmable. The factor which makes one a forward FFT and the other an inverse FFT is the difference in control signals to each stage of the FFT. Figure 91 shows a typical FFT stage with its corresponding control inputs. The FS signal into the FFT memory causes the split in the input apertures which corresponds to a butterfly diagram. The length control into the FFT memory changes the effective length of the shift registers in the memory. The only other inputs are the sin/cos references to the complex multiplier in the arithmetic section of the stage.

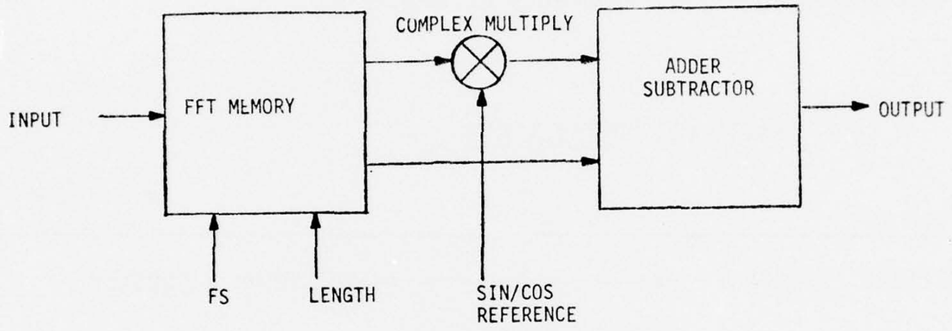


FIGURE 91. FFT STAGE

The control system for the FFT's need only supply the FS and sin/cos references to each stage at the proper time. The length controls are hard wired at the physical stage location. The same basic approach is taken in both the forward and inverse FFT control systems. The FS and sin/cos references are stored in PROM's. There is a reference set for each stage and all the FS signals are stored in a single PROM set. The control PROM containing the FS bits and also the'sin/cos reference sign bit are addressed from a direct address bus which comes from a counter. In the inverse FFT, there is a single sequence of bits stored in the PROM since the FFT timing does not change with mode changes. The forward FFT control PROM, however, has 3 distinct sections containing different sequences because the timing changes when a mode change is effected. The different areas of the PROM are accessed by using the mode control bits as part of the address. In the inverse FFT, the sin/cos references are also addressed by the direct address bus. However, each PROM has a data sequence which is unique to the stage for which it is associated. In the forward FFT, the reference PROMs cannot be addressed directly from a counter because not only does the timing change for a mode change but so does the actual data sequence. For this reason, an indirect address PROM lies between the direct address and the reference PROM. The indirect address PROM contains the addresses of the different sequences to be accessed in the reference PROM. The different indirect address sequences are accessed by the direct address and mode control bits. All the forward FFT reference PROMs contain identical data. The direct address for the forward FFT is obtained from the input buffer control counter. There is an address counter in the inverse FFT, however, which is synced from the sync PROM in the reorder memory control system.

7.5 OUTPUT BUFFER

The coefficients out of the inverse FFT are ordered in time with C_0 to $C_{N/2}$ on the Σ channel and $C_{(N/2+1)}$ to C_{N-1} on the Δ channel. N is the total samples in the aperture. The coefficients are ordered in frequency as shown in Figure 92a. The first sample out on the Σ channel is DC and increases in frequency to f_{MAX}. The first sample out on the Δ channel is f_{MAX} and decreases to DC. The region between sample K on the Σ channel and N-K-1 on the Δ channel in Figure 92b is the guard band. Thus only the first K samples of the Σ channel and the last K samples of the Δ channel are to be taken out of the output buffer. In addition, they are to be reordered as shown in Figure 92c. The output waveform starts at the most negative frequency at sample N-K-1 and increases through DC and up to the highest positive frequency at sample K.

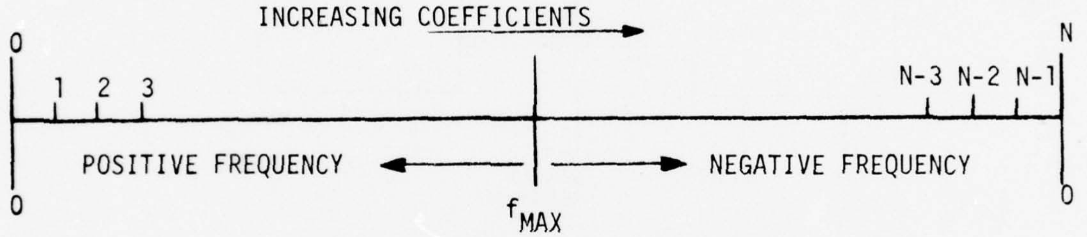


FIGURE 92a. APERTURE COEFFICIENT TO FREQUENCY RELATION

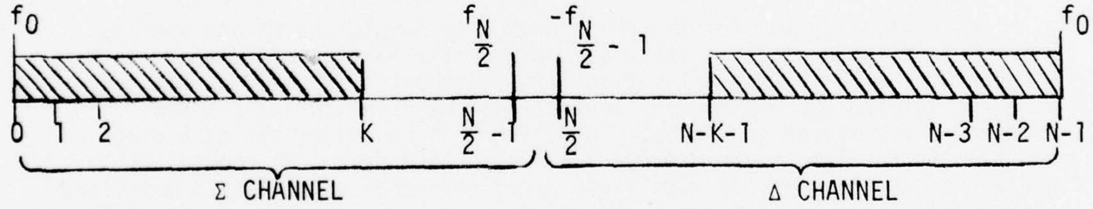


FIGURE 92b. DATA STORED IN OUTPUT BUFFER

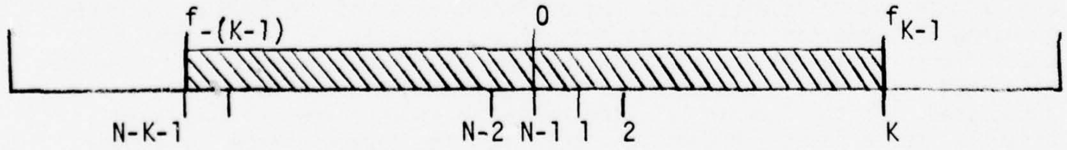


FIGURE 92c. DATA OUT OF OUTPUT BUFFER

A block diagram of the output buffer is shown in Figure 93a. In order to reorder the aperture as described above, an entire aperture must be stored before it is read out. This requires a double buffer architecture where one side is loading while the other side is dumping. The side that is loading is clocked with a gated process frequency and the side that is dumping is clocked with a gated sample frequency. The registers shown in Figure 93a are GUA registers. The A and B switches are the output switches on the GUA chip. The C switch is a control module switch function.

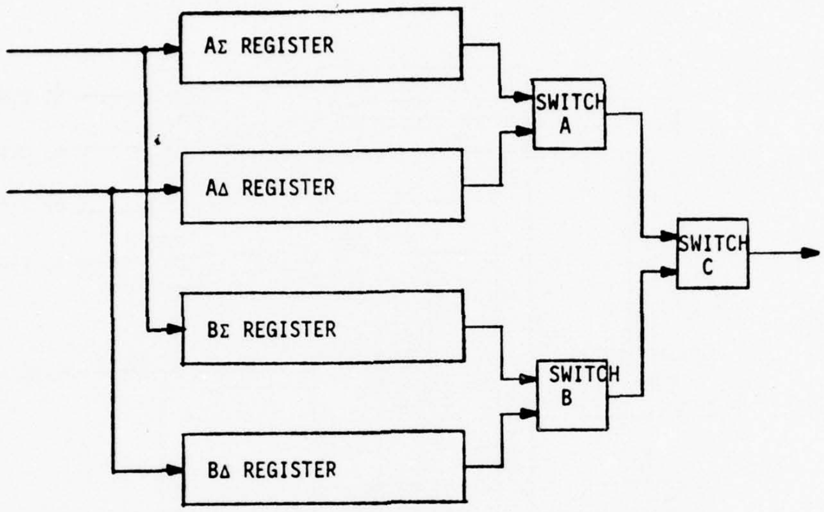


FIGURE 93a. OUTPUT BUFFER

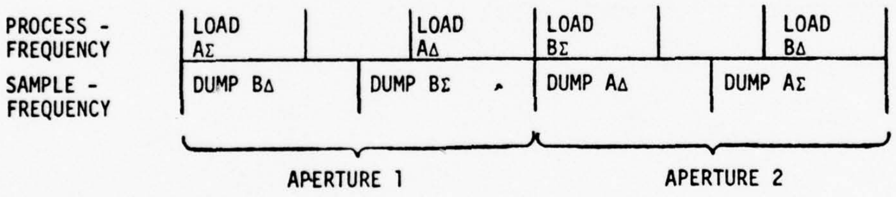


FIGURE 93b. OUTPUT BUFFER OPERATIONS

Figure 93b shows how the buffer operates. The AΣ register loads by turning on the process clock for the first K samples of the aperture and the AΔ register is loaded by turning on the process clock for the last K samples of the aperture. During the period of time that the A register set is being loaded, the B register set is being dumped. First the BΔ register is dumped, since it contains the negative frequencies, by applying the sample clock to it for K pulses. Then the positive frequency BΣ register is dumped by applying the sample clock for K pulses. By the time the B register set is dumped, the A register set is loaded. At this point, the A and B register sets switch functions. The output switches always point to that register which is dumping.

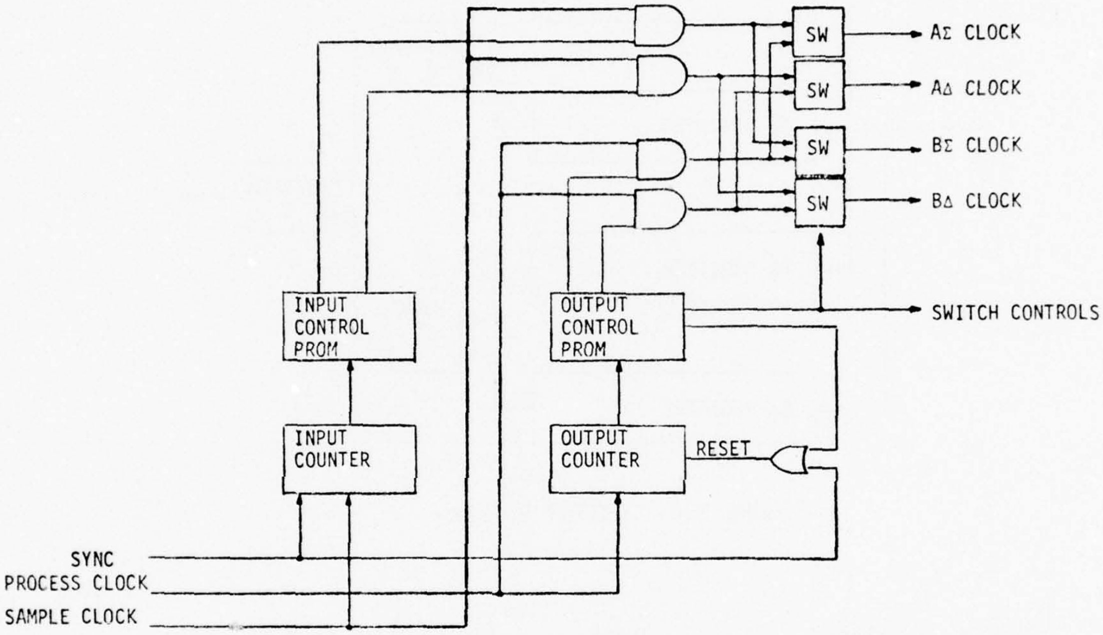


FIGURE 94. OUTPUT BUFFER CONTROL SYSTEM

A block diagram of the control system is shown in Figure 94. The two counters count the samples over a two aperture period. One is clocked at the process rate and the other at the sample rate. The counter being clocked at the process rate addresses the input control PROM which contains control bits for gating the process clock. The sample rate counter addresses the output control PROM which contains bits for gating the sample clock, switching the output switches, and resetting the counter. In addition to addressing the output control PROM, the sample counter also addresses the ramping and amplitude correction PROM before the square root approximator.

7.6 CLOCK GENERATOR

The clock generator produces the dual phase dual frequency clock required by the PWP system. The generator is divided into three sections.

1. Process frequency generator.

2. Input buffer sample frequency generator.

3. Output buffer programmable sync sample frequency generator.

A block diagram of the basic frequency generators is shown in Figure 95. The process frequency is generated by a simple voltage controlled oscillator (VCO) with a variable RC control network to obtain different process frequencies. The input buffer sample frequency is synthesized from the process frequency using the phase locked loop 1(PLL1) in Figure 95. The sample frequency has the following relationship to the process frequency:

$$f_S = 7/8\ f_P$$

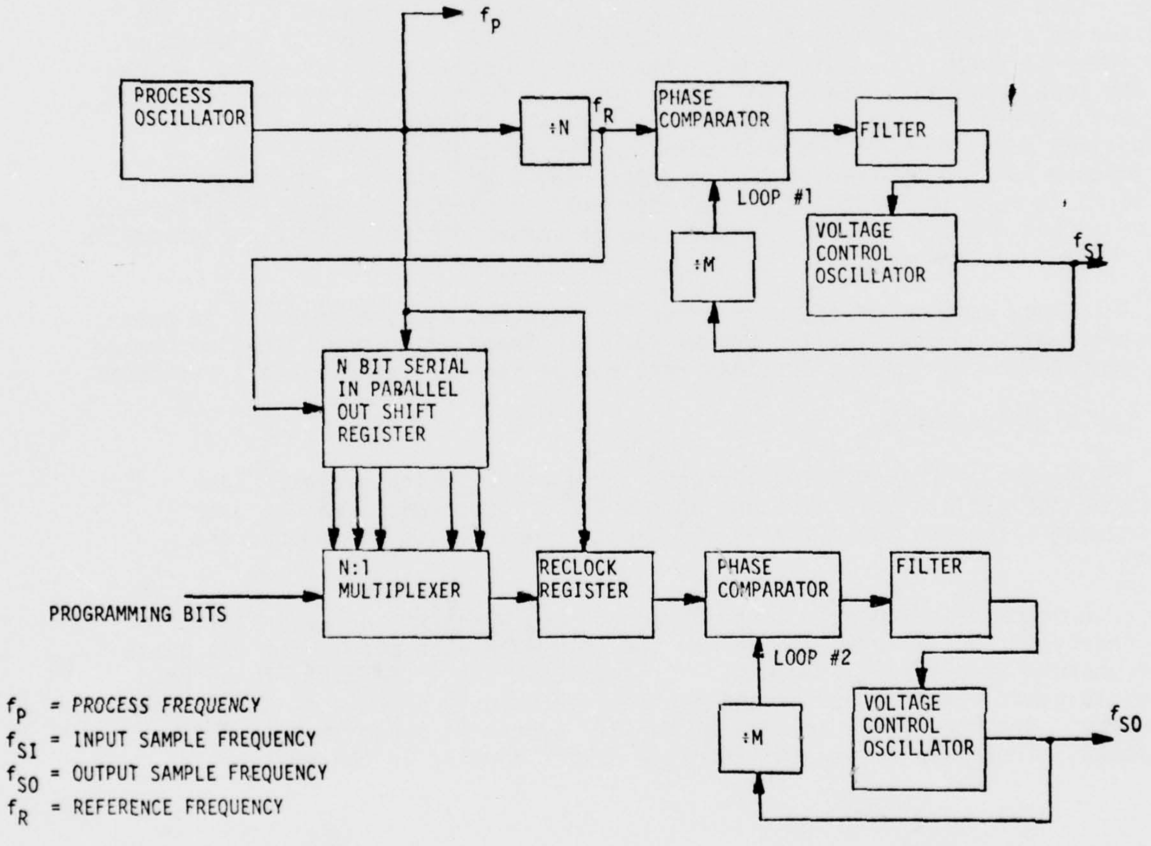

f_P
PROCESS OSCILLATOR
÷N
f_R
PHASE COMPARATOR
FILTER
LOOP #1
÷M
VOLTAGE CONTROL OSCILLATOR
f_{SI}
N BIT SERIAL IN PARALLEL OUT SHIFT REGISTER
PROGRAMMING BITS
N:1 MULTIPLEXER
RECLOCK REGISTER
PHASE COMPARATOR
FILTER
LOOP #2
÷M
VOLTAGE CONTROL OSCILLATOR
f_{SO}
f_P = PROCESS FREQUENCY
f_{SI} = INPUT SAMPLE FREQUENCY
f_{SO} = OUTPUT SAMPLE FREQUENCY
f_R = REFERENCE FREQUENCY

FIGURE 95. SYSTEM CLOCK GENERATOR

A reference frequency, f_R, is obtained for the loop by dividing the process frequency by 8. The 7/8 ratio is obtained by dividing the loop 1 output, f_{SI}, by 7 and feeding this to the loop phase comparator. The phase comparator will output a voltage which is proportional to the difference between f_R and $f_{SI}/7$. The comparator output is filtered and fed to a VCO which produces f_{SI}. The two frequencies f_P and f_{SI} are exactly in phase, their leading edges rise together, when f_R makes a low to high transition. This is important because the input and output buffer clocks can only be switched at this point. Consequently, the transition of f_R is used to sync the system. This places the start of an aperture at the input at the in phase point of the clocks.

A problem is encountered, however, with the output buffer sample frequency. Since the PWP is programmable and the pipeline length changes by a non-integral number of reference pulses, then for each operating mode, a new sync point must be defined with respect to the start of an aperture at the input of the output buffer. A new sync point can be defined for the output buffer phase locked loop (PLL2) by delaying the reference from one to eight pulses of the process clock. This is accomplished by using an eight bit serial in parallel out shift register as a delay line for f_R and by tapping off the right delay by means of an 8:1 multiplexer. The multiplexer output is reclocked with a register which has the same propagation delay as the divide by 8 counter used as the reference frequency generator. This eliminates the skew which would result from the propagation delay through the multiplexer. The reclocked output of the multiplexer is then applied to the reference input of loop 2. This loop is identical to PLL1 and produces a frequency which is 7/8 f_P. The only difference in the output of PLL2 and PLL1 is that the in-phase point of PLL2 is programmable with respect to that of PLL1.

The three outputs of the oscillator f_P, f_{SI}, and f_{SO} are then fed to phase splitters which produce the two phases for each input frequency. The two phases have adjustable overlapping up times required by the GUA dynamic shift registers.

7.7 CLOCK DISTRIBUTION

The clocks are distributed through a tree distribution system. Each output of the oscillator drives one or two TTL drivers which fan out over twisted pair lines to the clock driver modules distributed throughout the system.

Each clock driver module contains a receiver which drives several TTL to CMOS level shifter drivers. There are two inputs on each module for the phase 1 and phase 2 of a single frequency. A Texas Instruments SN75365 TTL to CMOS driver is used. Each output of the quad driver chip is capable of driving 68 pf at 10 MHz. The capacitive loading of the PWP system is approximately 4000 pf per phase. There are a total of 20 clock driver modules in the system.

SECTION VIII

PHYSICAL DESCRIPTION OF SYSTEM

8.1 MECHANICAL DESIGN

8.1.1 Overall Description

A straightforward mechanical design and layout of the PWP was used for ease of operation and test of the hardware. The general configuration is shown in Figure 96. It features a single backplane construction with a set of cooling fans mounted at one end. Channels for air flow are formed by the modules in a manner permitting air to pass directly over the components. The top edge of each module forms the closure for the air path. This permits individual modules to be removed or placed on extender cards while the system is in operation. Very little cooling efficiency is lost with the removal of one or two modules.

The summary data in Figure 96 covers the total PWP hardware which essentially fills the four-nest configuration. A total of 1745 integrated circuits are employed giving a component density of 1560 IC's per cubic foot including cooling, but excluding power supplies.

The modules in the PWP are mounted on 0.3 in. centers although the modules are designed for a potential of 0.2 in. centers if the narrower connector were available. The 0.2 in. spacing would give a component density of 2340 IC's per cubic foot and reduce the PWP volume from 1.12 to 0.75 cubic feet.

8.1.2 Thermal Control

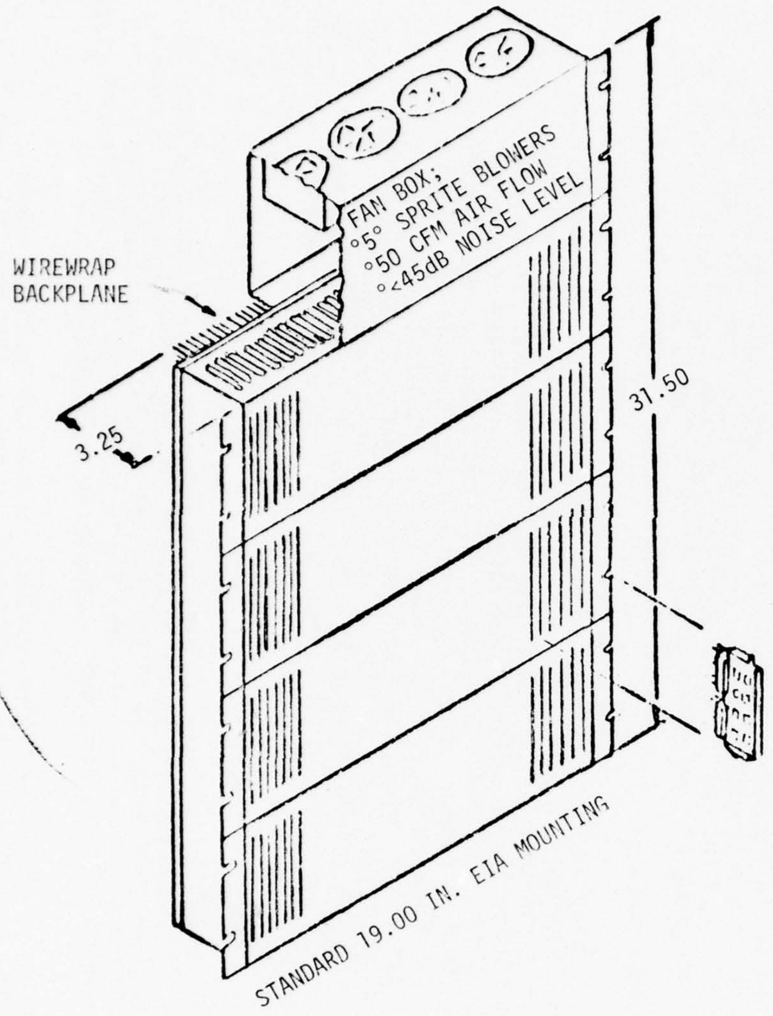
The estimated maximum dissipation for all of the modules is about 460 watts. The highest dissipation module is for the FFT memory at about 3.5 watts. A blower unit is provided that fits on top of the nest assembly. It contains 5 small fans that provide a uniformly distributed air flow through the module air passage slots. The fans are low speed quiet units which produce less than a total of 45 dB of audible noise. At the anticipated pressure drop through the nests, the fan assembly will pull about 50 CFM. With this air flow and heat dissipation, the temperature rise through the nests will not exceed 10°C. The efficiency of this method of cooling limits the worst case junction temperature rise above ambient to about 15°C. Since it is assumed that this equipment will be operated in room ambients of 25°C, the maximum junction temperatures will be at about 50°C.

8.2 NEST - BACKPLANE ASSEMBLY

The layout of the functional modules in the next-backplane assembly is shown in Figure 97. The space occupied by the forward and inverse FFTs and control is outlined. Figure 98 is a photograph of the FFT's in the nest. A photograph of the completed hardware together with the test bed is indicated in Figure 99. The PDP-11/20 computer used for all simulation and testing is in the background.

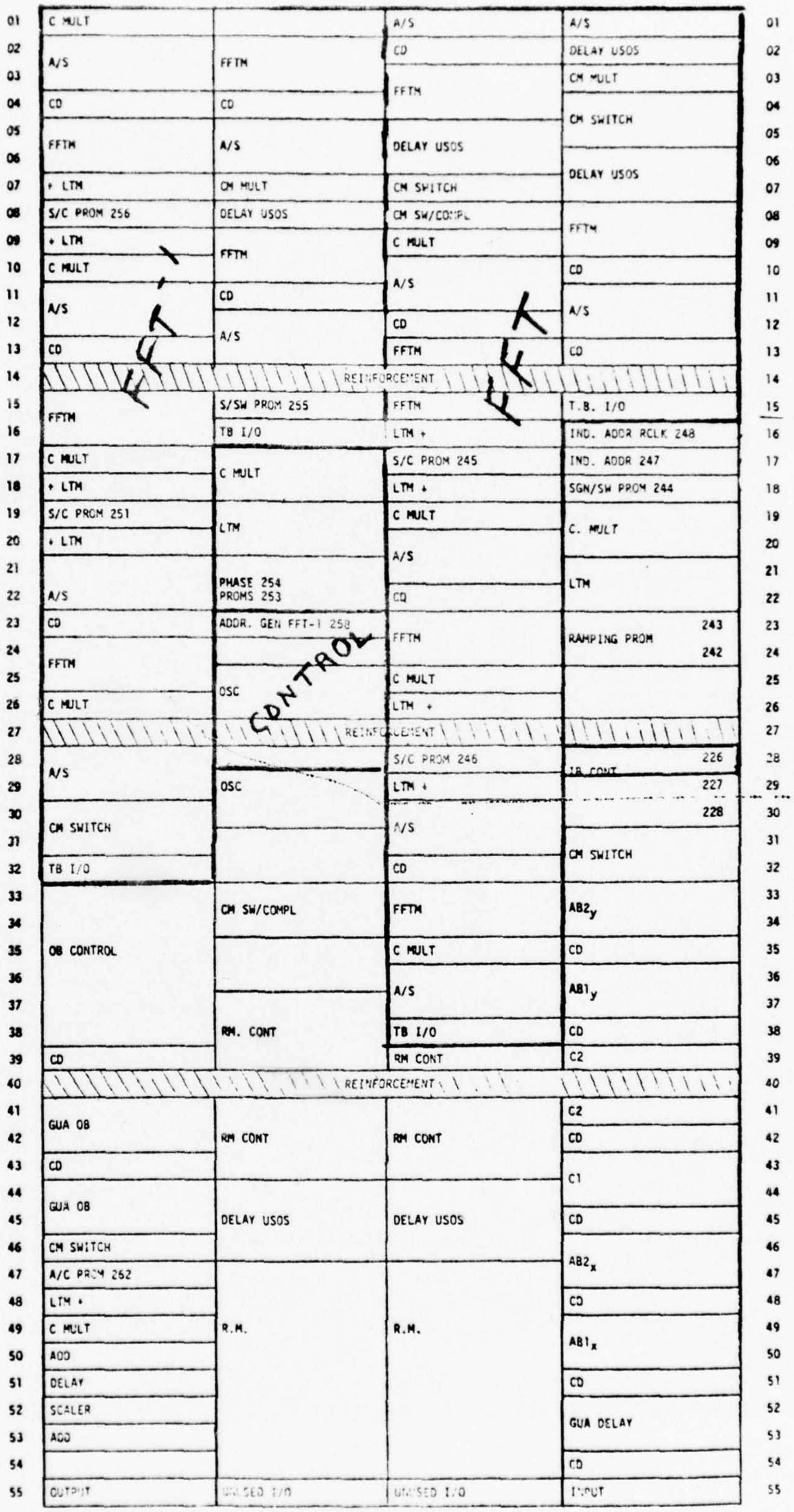
The extended cards with cables attached are test bed input and output probes. These have data registers incorporated on them so that test data can be inserted into the processor or picked off at maximum clock speed. The panel below the four nests is the 5 volt power supply.

It was necessary to design a special module extraction tool for removal of modules plugged into the nests. This is shown in Figure 100.



° TOTAL NO. OF MODULES	192
° NO. OF MODULE TYPES	8
° TOTAL NO. OF CIRCUITS	1745
° NO. OF CUSTOM SOS DEVICE TYPES	8
° TOTAL POWER DISSIPATION	460W
° AVE. DISSIPATION PER MOD.	2.39W
° MAX. MODULE DISSIPATION	3.5W
° POWER DENSITY	.23W/INS
° COMPONENT DENSITY	1560 IC'S/FT^3

FIGURE 96. PWP NEST CONFIGURATION



01	C MULT		A/S	A/S	01
02	A/S	FFTM	CD	DELAY USOS	02
03			FFTM	CM MULT	03
04	CD	CD		CM SWITCH	04
05	FFTM	A/S	DELAY USOS		05
06				DELAY USOS	06
07	+ LTM	CM MULT	CM SWITCH		07
08	S/C PROM 256	DELAY USOS	CM SW/COMPL	FFTM	08
09	+ LTM	FFTM	C MULT		09
10	C MULT		A/S	CD	10
11	A/S	CD		A/S	11
12		A/S	CD		12
13	CD		FFTM	CD	13
14	REINFORCEMENT				14
15	FFTM	S/SW PROM 255	FFTM	T.B. I/O	15
16		TB I/O	LTM +	IND. ADDR RCLK 248	16
17	C MULT	C MULT	S/C PROM 245	IND. ADDR 247	17
18	+ LTM		LTM +	SGN/SW PROM 244	18
19	S/C PROM 251	LTM	C MULT	C. MULT	19
20	+ LTM		A/S		20
21	A/S	PHASE 254 PROMS 253		LTM	21
22			CD		22
23	CD	ADDR. GEN FFT-1 258	FFTM	RAMPING PROM 243	23
24	FFTM			242	24
25		OSC	C MULT		25
26	C MULT		LTM +		26
27	REINFORCEMENT				27
28	A/S		S/C PROM 246	226 IB CONT	28
29		OSC	LTM +	227	29
30	CM SWITCH		A/S	228	30
31				CM SWITCH	31
32	TB I/O		CD		32
33	OB CONTROL	CM SW/COMPL	FFTM	$AB2_y$	33
34					34
35			C MULT	CD	35
36			A/S	$AB1_y$	36
37		RM. CONT			37
38			TB I/O	CD	38
39	CD		RM CONT	C2	39
40	REINFORCEMENT				40
41	GUA OB	RM CONT	RM CONT	C2	41
42				CD	42
43	CD			C1	43
44	GUA OB	DELAY USOS	DELAY USOS		44
45				CD	45
46	CM SWITCH			$AB2_x$	46
47	A/C PROM 262	R.M.	R.M.		47
48	LTM +			CD	48
49	C MULT			$AB1_x$	49
50	ADD				50
51	DELAY			CD	51
52	SCALER			GUA DELAY	52
53	ADD				53
54				CD	54
55	OUTPUT	UNUSED I/O	UNUSED I/O	INPUT	55

FIGURE 97. PWP BACKPLANE LAYOUT

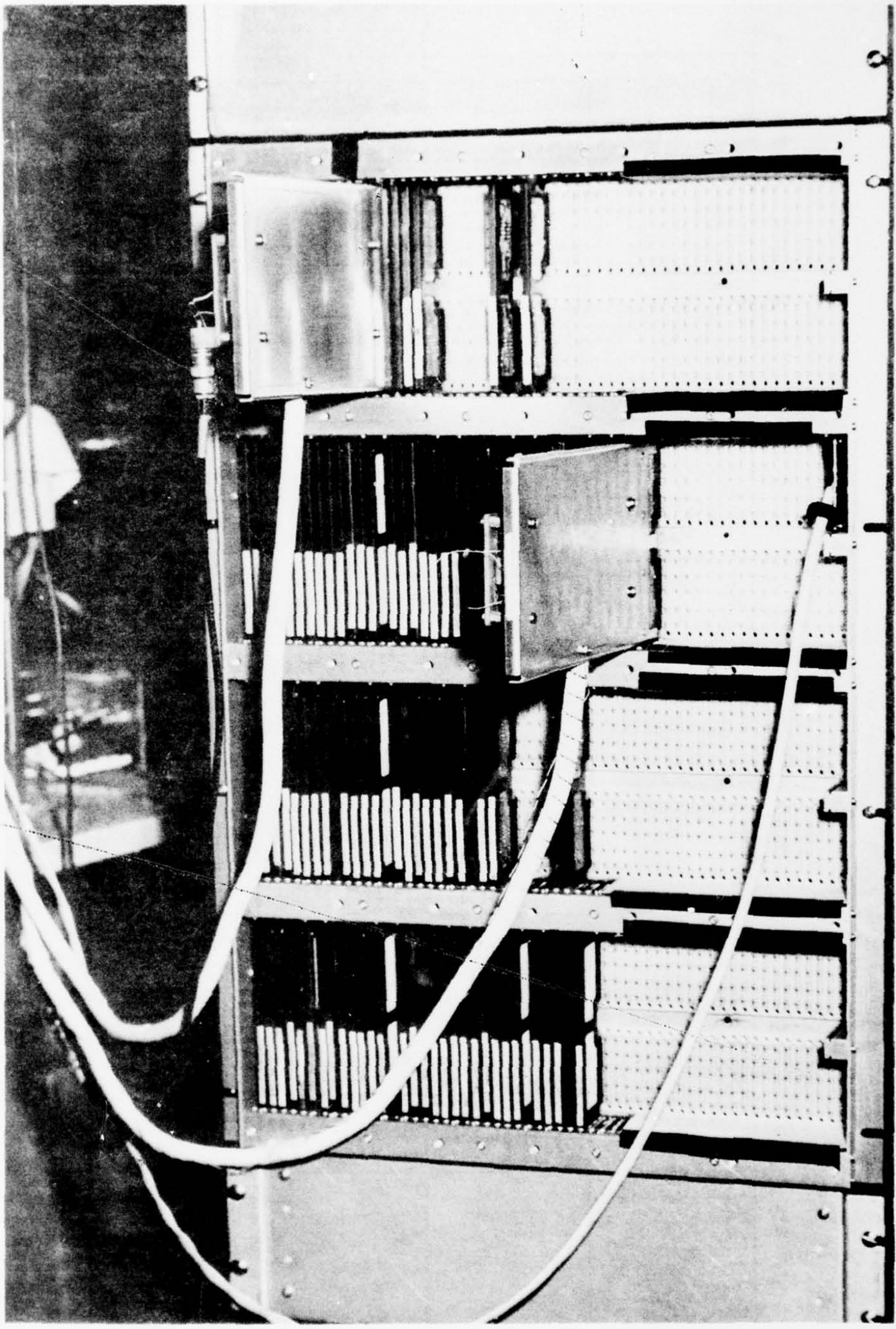


FIGURE 98. FORWARD AND INVERSE FFT'S IN PWP NEST

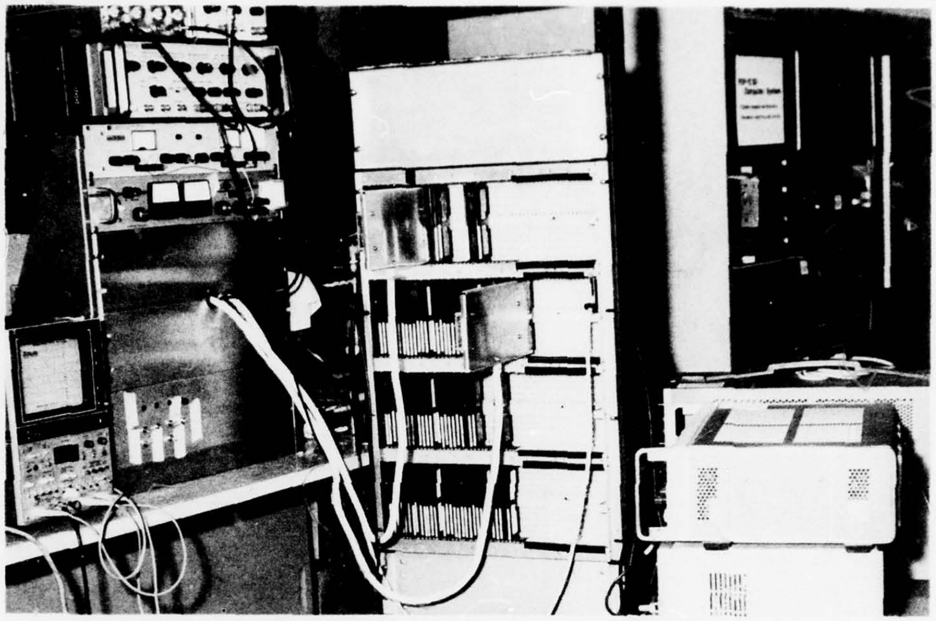


FIGURE 99. PWP FFT'S WITH TEST BED AND PDP-11/20 COMPUTER

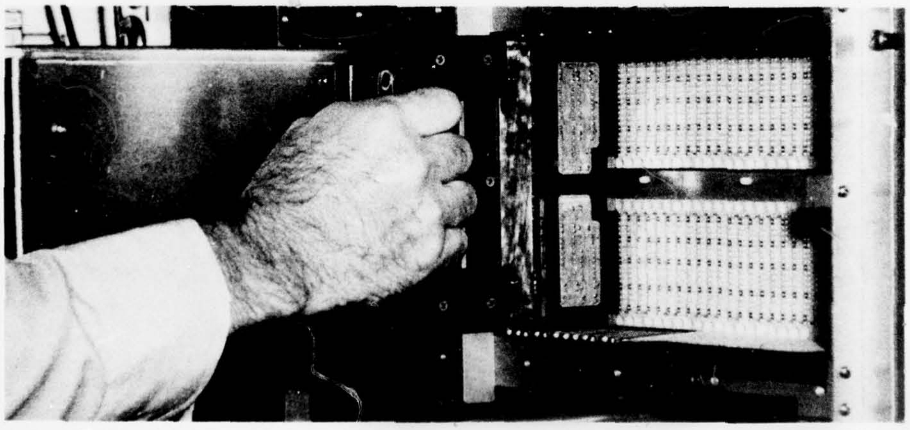


FIGURE 100. MODULE EXTRACTION

PrecedING Page BLank - NOT FiLMed

SECTION IX

SYSTEM SOFTWARE DEVELOPMENTS

9.1 PWP SYSTEM SIMULATION

The PWP system performance simulations were presented in detail previously (1, 2) and will only be summarized here. The objectives of the simulation efforts were to verify the step transform algorithm and determine its performance level as a function of various hardware design choices particularly the number of quantization bits. Simulations were conducted for all five time-bandwidth products and two SAR cases.

The spectrum of a linear FM pulse with a rectangular time envelope is not exactly rectangular in amplitude nor is its phase exactly a linear group delay. This deviation from the ideal is especially severe for small time bandwidth products, therefore, the use of -40 dB weighting on a small time bandwidth product waveform fails to achieve a -40 dB sidelobe level(10). This fact accounts for the two weightings of -40 dB and -35 dB used in the step transform filter. Sidelobes of -40 dB are readily achieved for the higher time bandwidth products (>295), however, -40 dB sidelobes require special weighting function design with the smaller time bandwidth products, therefore a -35 dB weighting has been selected as a practical sidelobe level for those waveforms with WT $\leq$ 147.87.

The processor performance with noise was determined by using a single point target as input, adding a known noise, and finding the output signal to noise (S/N). Measurements were made as the range of the target was varied. The measurements of output signal-to-noise ratio were within 0.1 dB of the theoretical values for -35 dB and -40 dB Taylor weighting.

Integrated sidelobe level measurements were made by finding the ratio of the sum of the squares of the values constituting the main lobe to the sum of the squares of the values comprising the remainder of the compressed pulse. Table 33 shows the results of the integrated sidelobe measurements for all time-bandwidth cases as a function of quantization level. The results for the PWP system are also shown, and all bit values include sign. An interesting observation from Table 33 is that the integrated sidelobe levels for the system are less than -25 dB for time bandwidth products $\geq$ 295.8, and that there is very little increase in integrated sidelobe levels as the number of bits is decreased from 13 to the system configuration. No appreciable deterioration due to quantization is observed until the word size is reduced to 7 bits (6 bits plus sign).

Simulations run in studying PWP performance included both an analytical model for waveform generation and the waveform generator hardware simulator. There was no discernible difference between the analytic waveform case and the PWP hardware case for TW products less than 1183 and even in this case the differences were only observable in the sidelobe levels below -55 dB.

Another function of the PWP, in addition to range pulse compression, is compression of azimuth samples for fine azimuth resolution. This mode of operation uses three time bandwidth products and acts as a matched filter only since no waveform generation is involved. The first case is WT = 73.9 which is the identical configuration for the range pulse compression of the same WT. In addition to the WT = 73.9 case, two other cases were studied for azimuth "focusing". The focusing cases are configured by varying the slope of the waveform reference. In all cases the azimuth sodelobes are maintained at the weighting level of -35 dB.

TABLE 33
INTEGRATED SIDELOBE LEVELS

WT PRODUCT	AVERAGE INTEGRATED SIDELOBE LEVEL (dB)				
	13 BIT	11 BITS	9 BITS	7 BITS	PWP SYSTEM
73.9	-21.38	-21.35	-21.32	-20.66	-21.22
147.9	-22.45	-22.43	-22.31	-21.13	-22.37
295.8	-26.41	-26.39	-26.18	-24.06	-26.11
591.5	-28.02	-27.99	-27.59	-24.69	-27.51
1183.0	-26.81	-26.76	-26.36	-23.6	-26.36

9.2 WAVEFORM GENERATION MODE SOFTWARE

A great deal of effort was expended in the development of the waveform generation software. Several problems were encountered which were extremely difficult to solve. The three major problems encountered were:

1. Originally two separate simulators were written. One was for the pulse compression mode and the other for the waveform generation. The two were not similar. In order to have a simulator perform both pulse compression and waveform generation as the hardware would, it was necessary to reconfigure the pulse compression simulator to perform both functions.

2. The pulse compression simulator simulated the hardware functions but not the final hardware structure. It was necessary to restructure the original simulator to match the hardware as close as possible. This was necessary not only for the waveform generation but also the pulse compression in order to extract the reference data to be contained in PROM's in the hardware.

3. The reference information was not analytically derived for the pulse compression simulator and there was therefore no analytical means of obtaining the waveform compression reference data from the pulse compression references.

The solution to the first problem did not so much involve a physical change of the pulse compression simulator but understanding how to stimulate the system input to produce the proper input to the inverse FFT for waveform generation. It was previously known what that input to the inverse FFT should be because of the original waveform generation simulator. However, a stimulation at the system input is transformed by the forward FFT amd reordered in the reorder memory. It was found that an impulse at the input would produce inversely ordered coefficients at the reorder memory output and that these coefficients carried the wrong phase terms. To correct this situation is the subject of the third problem and proved to be extremely difficult.

The second problem applied to both pulse compression and waveform generation. The major difficulty here was in the fact that the phase correction in the simulator took place prior to the reorder memory and prior to the PAK subroutine. The PAK subroutine corrects the difference between the forward FFT simulator output coefficient order and the coefficient order required by the algorithm. The hardware was designed to have the phase correction occur after the reorder memory. This difference involved translating the phase references through the PAK subroutine and the reorder memory and then rewriting the phase generation program to generate the new phases in the required sequence. The algorithm was then changed to perform the multiplication after the reorder memory and the new algorithm was verified in the pulse compression mode by cross checking the response with the equivalent response of the old algorithm.

The third problem was by far the most difficult and time consuming. By stimulating the PWP input with an impulse, it was possible to obtain impulses in each aperture of the reorder memory output. However, each impulse had a phase term associated with it from the forward FFT and was in decreasing rather than increasing frequency order. Since there was no analytical basis for calculating the proper phases and weighting, the output of the new waveform generator had to be compared to the output of the original simulator and the phase differences divided out. The output was then ramped down instead of up and the complex conjugate taken to obtain an increasing linear FM waveform. An analysis program was written which would take the waveform generator output and divide the phase difference out of each aperture and detect a common phase difference between all apertures due to the ramping waveform. The difference due to the phase correction prior to the inverse FFT was then applied to the phase reference terms. The ramping difference was left alone until zero phase difference due to the phase correction term was detected. This required a repetitive procedure in which a correction was made, the program rerun, and a new correction made. Each pass through the program improved the response until finally there was no difference between the original simulation and the new simulation except for the fixed phase difference due to the ramping. At this point, the program corrected the output ramping waveform by dividing out the fixed phase difference which was due to misalignment of the ends of the ramp with the phase of the aperture. The result was a match to four decimal places with the original simulation. The program run spanned three passes in 30 minutes for a 16x16 configuration to 16 passes in 8 hours for a 64x64 configuration.

9.3 REORDER MEMORY

A functional model of the reorder memory was simulated in order to verify the hardware configuration and to obtain the base address sequence required for the addressing algorithm.

The model used coefficient numbers instead of actual data in order to track individual words through the memory and observe data patterns in the memory. A bit reversed coefficient sequence was generated to simulate the coefficients leaving the forward FFT on Σ and Δ channels. An unordered coefficient matrix was then generated for a full waveform and diagonalized. Since the data is read out of the memory in order, then the bit reversed order along the diagonal corresponds to the coefficient number in ordered sequence to be read out of the memory. By locating the coefficients in bit reversed order along the diagonal in ordered sequence in the memory and then reading the addresses of the bit reversed input sequence in aperture order, the address sequence is obtained for the entire memory. By studying the address pattern, it became apparent that the address sequence is simply a base sequence with increasing multiples of a constant added for each successive aperture. The program then used this sequence as a basis for address generation and printed the output sequence which was verified as the bit reversed sequence along the diagonal of the ordered coefficient matrix. The tables of base addresses for each of the five PWP system configurations are given below.

16 x 16
PROM Recorded Base Address Sequence
Σ - Channel

Count	Memory	Address
1	AX	1
2	AY	25
3	BX	29
4	BY	21
5	AX	0
6	AY	24
7	BX	28
8	BY	20

Subsequent addresses are generaged by adding increasing multiples of 2 to the above sequence.

16 x 32
PROM Recorded Base Address Sequence
Σ - Channel

Count	Memory	Address
1	AX	1
2	AY	49
3	BX	57
4	BY	41
5	AX	62
6	AY	46
7	BX	54
8	BY	38

Subsequent addresses are generated by adding increasing multiples of 2 to the above sequence.

32 x 32
PROM Recorded Base Address Sequence
Σ - Channel

Count	Memory	Address
1	AX	1
2	AY	97
3	BX	113
4	BY	81
5	AZ	122
6	AY	90
7	BX	106
8	BY	74
9	AX	127
10	AY	95
11	BX	111
12	BY	79
13	AX	120
14	AY	88
15	BX	104
16	BY	72

Subsequent addresses are generated by adding increasing multiplies of 4 to the above address sequence.

32 x 64
PROM Recorded Base Address Sequence
Σ - Channel

Count	Memory	Address
1	AX	1
2	AY	193
3	BX	225
4	BY	161
5	AX	242
6	AY	178
7	BX	210
8	BY	146
9	AX	251
10	AY	187
11	BX	219
12	BY	155
13	AX	236
14	AY	172
15	BX	204
16	BY	140

Subsequent addresses are generated by adding increasing multiples of 4 to the above sequence.

64 x 64
PROM Recorded Base Address Sequence
Σ - Channel

Count	Memory	Address
1	AX	1
2	AY	385
3	BX	449
4	BY	321
5	AX	482
6	AY	354
7	BX	418
8	BY	290
9	AX	499
10	AY	371
11	BX	435
12	BY	307
13	AX	468
14	AY	340
15	BX	404
16	BY	276
17	AX	509
18	AY	381
19	BX	445
20	BY	317
21	AX	478
22	AY	350
23	BX	414
24	BY	286
25	AX	495
26	AY	367
27	BX	431
28	BY	301
29	AX	464
30	AY	336
31	BX	400
32	BY	272

Subsequent addresses are generated by adding increasing multiples of 8 to the above address sequence.

SECTION X

PWP SYSTEM TEST FACILITY

The PWP system test facility is a computer based multifunction system with a high degree of flexibility for operating and analyzing the PWP system. The test facilities design objectives are:

1. Allow the PWP system to be operated in both the pulse compression and waveform generation modes.

2. Allow the pulse compression mode operation with either the computer or PWP as the waveform source.

3. Allow subfunctions of the PWP system to be operated independently of the rest of the system.

4. Allow flexible probing of the PWP system on a module to module basis for trouble shooting purposes.

5. Provide a means for displaying a real time output of the PWP system.

The PWP test facility is shown in Figure 101. The facility consists of a PDP-11/20 computer system and a high speed configurable memory and control system. The computer system consists of the PDP-11/20 CPU with 28K of core memory, two RK05 high density disks and one RK03 low density disk, a TU-10 magnetic tape unit, a line printer and video terminal. The computer is used to generate input waveforms, simulate the PWP system or subsystem under test, perform data transfers to and from the high speed memory and load test conditions into the high speed memory control system, perform error analysis on the PWP output waveform, and display the PWP output waveform.

The high speed memory consists of forty four 1Kx1 TTL random access memories which are capable of read or write cycles in excess of 10 MHz. The memories can be multiplexed for 1Kx44, 2Kx22, or 4Kx11 configurations. The basic configurations are for the 44 bit data probe, PWP input/output buffers, and PDP-11 interface respectively. The memory is capable of reading out data and transmitting it to the PWP and then writing data back in from the PWP at 10 MHz or simultaneously reading and writing at 5 MHz. It can operate in a burst mode where the memory contents are dumped and then overwritten by the PWP output or in a continuous mode where the memory is continuously read but not overwritten.

The memory control system consists of computer interfacing hardware, PWP interfacing hardware, and memory addressing and multiplexing hardware and data normalizing circuits. The memory and controls are contained on two 9 inch x 9 inch Cambion wire wrap cards. The system consists of 230 standard and Schottky clamped TTL circuits with a total power dissipation of 60 watts.

10.1 PDP-11/20 COMPUTER SYSTEM

A PDP-11/20 computer with 28K 16 bit words of core memory was used for this job. The computer utilizes a disk operating system through which the user can call programs into core from disk or magnetic tape. The user interfaces

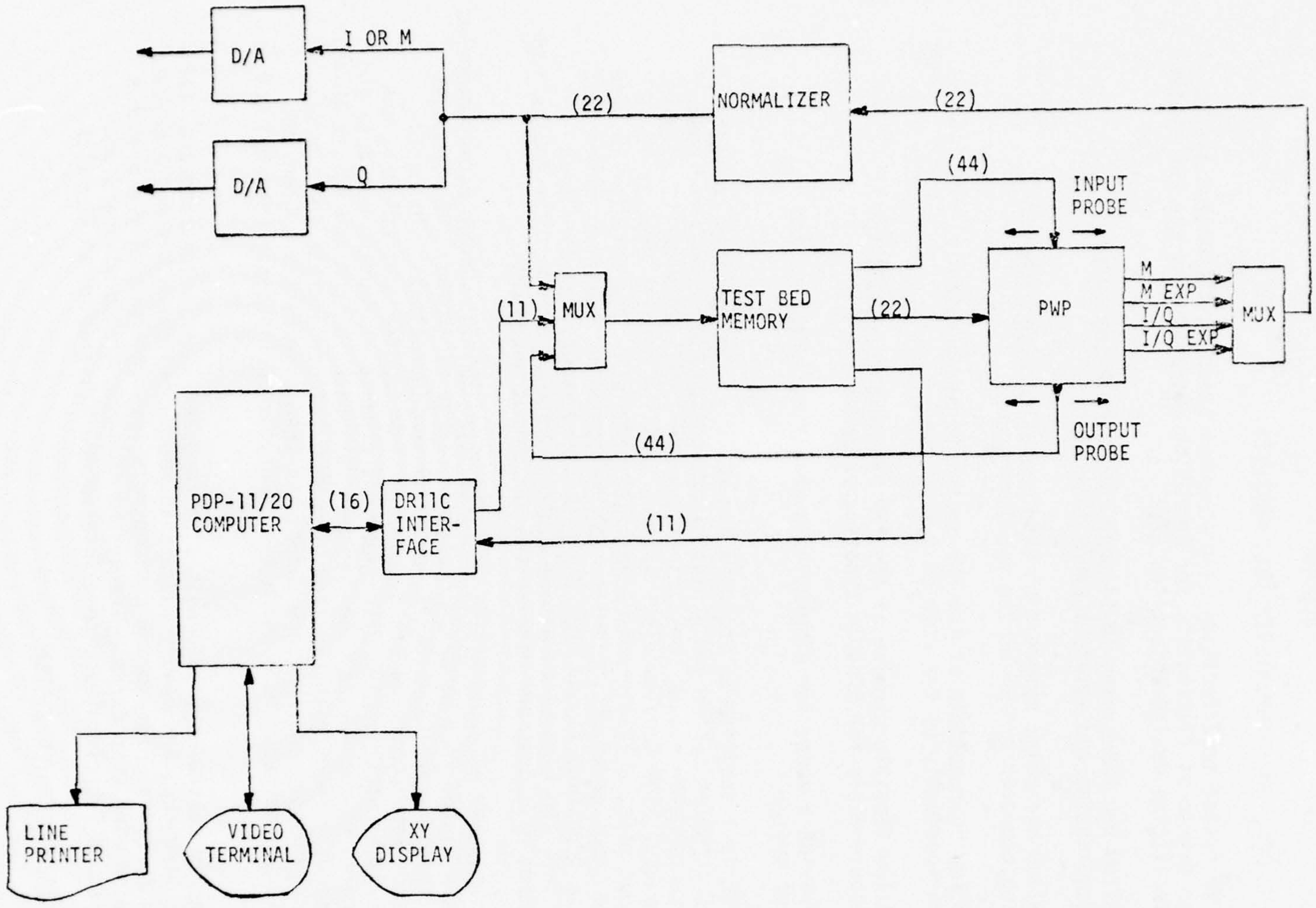


FIGURE 101. PWP SYSTEM TEST BED

with the computer through a video terminal. Programs and data can be printed out on either a line printer or the video terminal or through a parallel interface device. This latter device allows the computer to communicate with a user defined hardware system such as the high speed memory of the PWP test system.

Programs are input into the computer through the file utility program or the edit program. Programs can be written in either FORTRAN or assembly language. The FORTRAN or MACRO compilers create machine language programs called OBJECT modules which are then linked together as a unit called a LOAD module. The LOAD module is the actual program called into core by the DOS monitor when the user requests the computer to run a program.

Computer programs can be structured in two ways. The first is the standard main program/subroutine structure used in all computers. The main program calls on different subroutines to perform specific functions required. All subroutines are linked with the main program and loaded as a unit into core. The second structure is called an overlay structure and is a result of limited core space. Sometimes a program and its associated data arrays are too large to fit into core. In such cases, the program may be broken into functional units called overlays. An overlay consists of a main program and associated subroutines. However, when the program is run a program called the core resident program is loaded into core and it is this program which calls into core the various overlay programs associated with it. The distinction is that the overlays do not exist in core when the user loads the program as the subroutines do for the subroutine structured approach. The overlays reside on disk until they are called by the resident.

Very large data arrays are generated and manipulated by the PWP simulator and test software. These arrays consume core space at an enormous rate so that an alternative approach must be used for treating these arrays. In general, data files are created on the disk which are accessed by the program for data transmission. The data files used throughout the PWP software are formatted files which require contiguous disk space. For this reason, most data is stored on a separate disk from the system operating disk. The system operating disk randomly stores programs wherever it finds free space and contiguous disk space is generally at a premium. Disk space is defined in terms of blocks. A file block is 64 words and a disk block is 4 file blocks or 256 words. There are 3650 disk blocks available on a high density disk.

Several problems were encountered with the computer system during the course of the PWP program. The first was that the computer was reconfigured and a second high density disk added in the system disk position. This change required a new DOS monitor and system software. However, the version 9 system which we received had an error which prevented data transfers between overlays. By the time version 10 software was issued by Digital, a significant amount of software had been written for the module test bed.

A second problem was when a disk which contained the transmit mode phase analysis program and system development programs was physically dropped and dented making the programs unrecoverable. Several man months were lost re-deriving these programs.

In addition to these problems, several hardware failures occurred which took weeks to diagnose.

10.2 PWP SYSTEM TEST HARDWARE

The system test hardware consists of the high speed memory, the memory control system, a data normalizer, and computer and PWP logic interfacing hardware. Figure 102 shows a block diagram of the hardware system. The heart of the system is the 1Kx44 random access memory. The memory is designed to accept inputs from four sources:

1. PDP-11/20 computer via the DR11-C parallel interface module.

2. The 44 bit test probe from the PWP.

3. The normalized I and Q channel outputs of the PWP.

4. The normalized $\sqrt{I^2 + Q^2}$ output of the PWP.

In the diagram, M designates the $\sqrt{I^2 + Q^2}$ PWP output and I and Q the other outputs. The memory outputs data to one of three devices:

1. The PDP-11/20 computer via the DR11-C parallel interface.

2. The PWP input buffer.

3. The PWP input test probe.

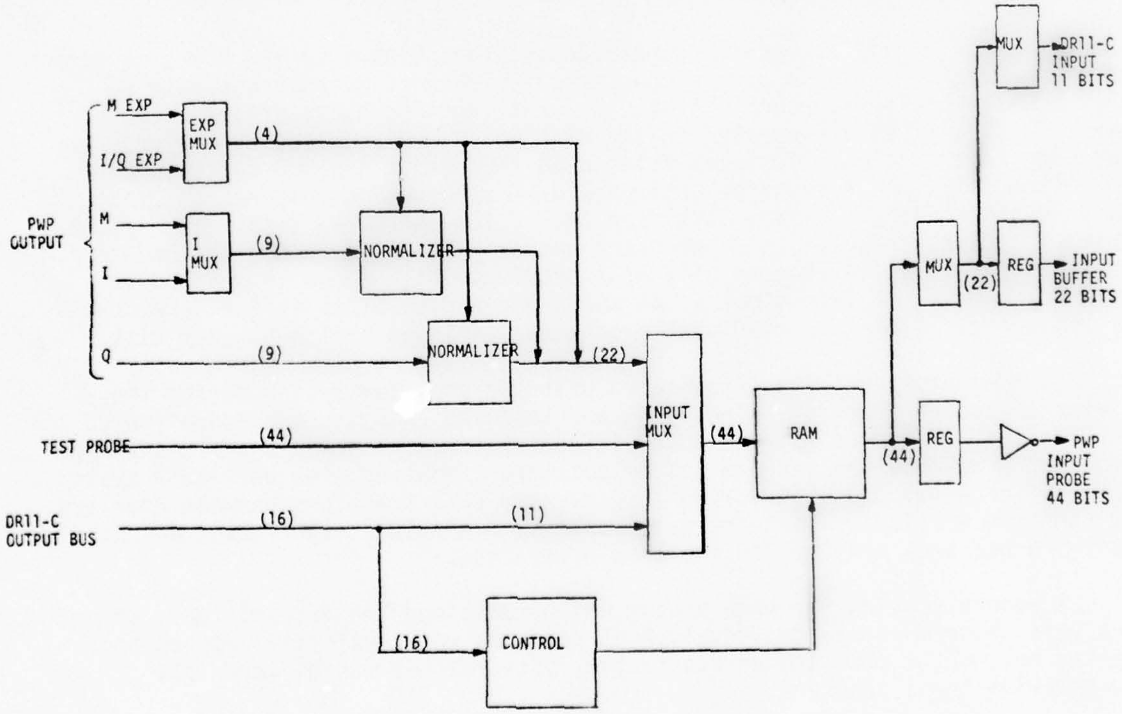


FIGURE 102. PWP TEST BED HARDWARE FUNCTIONS

The memory outputs to only one of the above devices at a time. In each of the above cases, the memory is configured in either 4Kx11 bits, 2Kx22 bits, or 1Kx44 bits.

A test is performed by first loading the test waveform into the memory and test parameters into the control system. The data is then passed at speed to the PWP system. The PWP system output waveform is then stored into the memory before being transmitted back to the computer. There are five functional tests designed into the memory control system.

10.2.1 Functional Design

The PWP test system was designed to perform five separate test functions:

1. Waveform generation where the PWP output is transmitted back to the computer for analysis.

2. Pulse compression where the waveform is computer generated and the number of samples in the waveform is less than 1024.

3. The same as 2 except where the number of samples are greater than 1024.

4. Pulse compression where the linear FM waveform is generated by the PWP, stored in the memory, and returned to the PWP.

5. Subfunction testing with the 44 bit test probes.

Figure 103 shows a functional block diagram of the waveform generation test. In this mode, the PWP generates its own impulse input upon a signal from the memory controls. The system is to operate at 10 MHz and the linear FM output of the PWP is to be stored in the I and Q format in the memory. The waveform is then transferred to the computer for analysis.

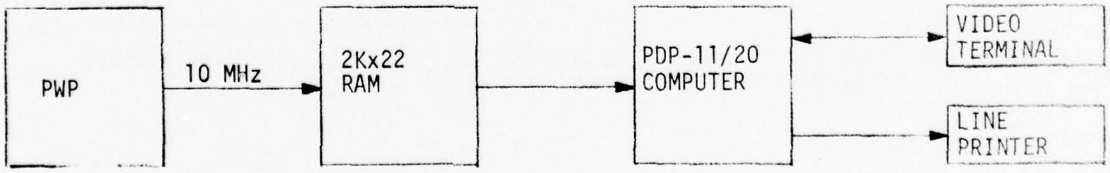


FIGURE 103. WAVEFORM GENERATION CONFIGURATION OF PWP TEST BED

Figure 104 shows a block diagram of the system in the pulse compression mode where the input waveform is computer generated. In this mode, the waveform which is less than 1024 samples is stored in one half of the 1Kx44 bit memory. This memory is read out to the PWP at the 10 MHz rate where it is processed and returned to the other half of the memory. The second memory may then

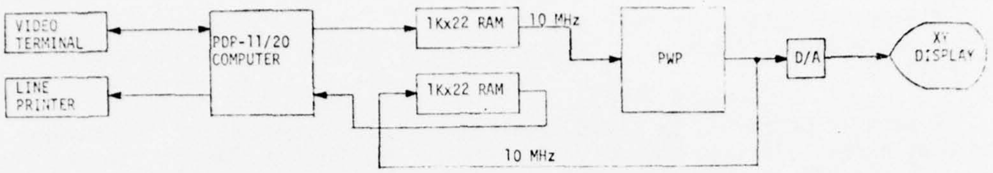


FIGURE 104. PWP TEST BED IN PULSE COMPRESSION CONFIGURATION WITH COMPUTER GENERATED WAVEFORM NUMBER OF SAMPLES < 1024

transfer the PWP output back to the computer. The advantage to this is that the input waveform is not destroyed by overwriting. The input waveform can be recycled through the PWP and displayed on an XY video display or oscilloscope.

Figure 105 shows the system in the pulse compression mode where the number of samples is greater than 1024. In this mode, both sides of the memory are required to store the input waveform. In this case, writing back to the memory destroys the original contents and prohibits real time display. This is not seen as much of a disadvantage, though, because the waveform can be displayed through the computer's D/A converter and XY display. There is no processing performed on the actual data in the computer.

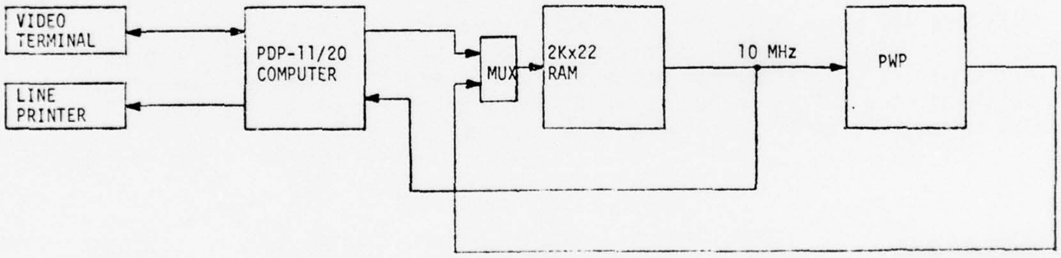


FIGURE 105. PWP TEST BED IN PULSE COMPRESSION CONFIGURATION WITH COMPUTER GENERATED WAVEFORM NUMBER OF SAMPLES > 1024

Figure 106 shows the system in the pulse compression mode where the input waveform is generated by the PWP. The linear FM is generated by the PWP and stored in the memory. The memory is then read out to the PWP and the processed waveform is stored again in the memory. This output waveform is then transferred to the computer for analysis.

Figure 107 shows the system in the subsystem probe configuration. In this mode, the memory is loaded with 44 bits of data from the computer. The data is then transferred to the PWP in one of three ways:

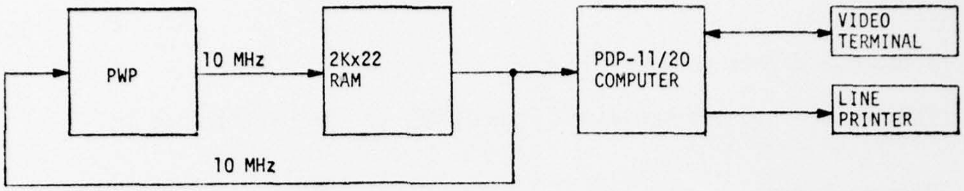


FIGURE 106. PWP TEST BED IN PULSE COMPRESSION CONFIGURATION WITH PWP GENERATED WAVEFORM

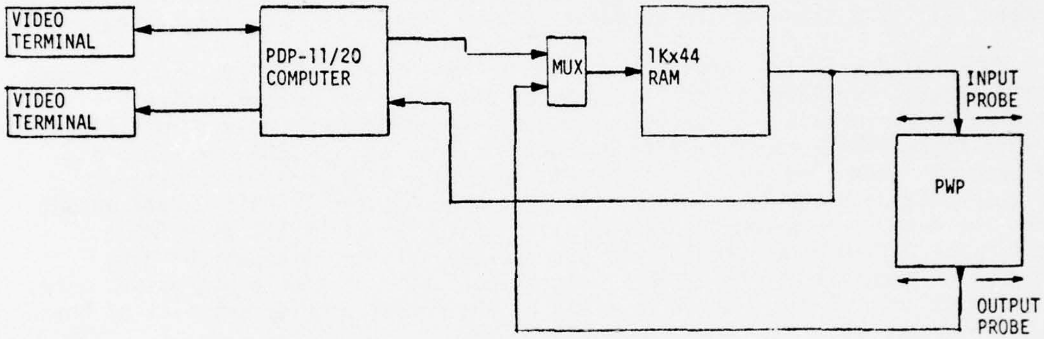


FIGURE 107. PWP TEST BED IN PROBE CONFIGURATION

1. The data is continuously read from the memory and cycled through the subsystem under test.

2. The memory is read once and overwritten with the test probe output at 5 MHz. The delay through the subsystem is less than the length of the input waveform.

3. The memory is read once and overwritten with the test probe output at 10 MHz. The delay through the subsystem is greater than the length of the input waveform.

The first method is used for running the subsystem in a cyclic manner so that the states of some intermediate point in the subsystem can be manually probed and observed by oscilloscope or logic state analyzer. This is done mostly for intermittant problems and finding wiring shorts. The other two are for computer analysis of the subsystem output and eliminates most of the labor of manual probing.

All five of the above test functions are designed into the test bed and wired on the wire wrap boards. However, only the last function, the subsystem probe configuration, was populated with circuits, since the FFT's are the only PWP subsystems implemented with SOS modules. The remaining four functions can be implemented by populating the balance of the wire wrap boards with the proper circuits.

10.2.2 Hardware and Interface Design

The PWP system test bed hardware (Figure 108) can be partitioned in five functional categories:

1. Control System
2. Data Directing
3. Data Normalizer
4. High Speed Memory
5. Data Transmission

The control system is responsible for interfacing with the PDP-11/20 computer and directing the progress of the test once the computer has given the START command. It is a slave to the computer via the DR11-C parallel interface.

The DR11-C parallel interface is tied to the PDP-11/20 UNIBUS and responds to the computer programs in the same way as any other peripheral device attached to the UNIBUS. It provides to the user hardware 16 data output lines, 16 data input lines, two user defined control lines called REQA and REQB, and two program controlled lines called CSR0 and CSR1. Data and control signals are transmitted to the test bed control system across the 16 DR11-C data output lines and data is returned to the computer across the 16 DR11-C data input lines. The DR11-C data output lines are divided into two fields; the data field which consists of the 12 least significant bits of the 16 bit output word, and the bus control field which consists of the 4 most significant bits of the 16 bit output word.

The control system consists of a number of data registers, two address registers, two address comparators, two address multiplexers and a control word decoder. The control system block diagram is shown in Figure 108. The control

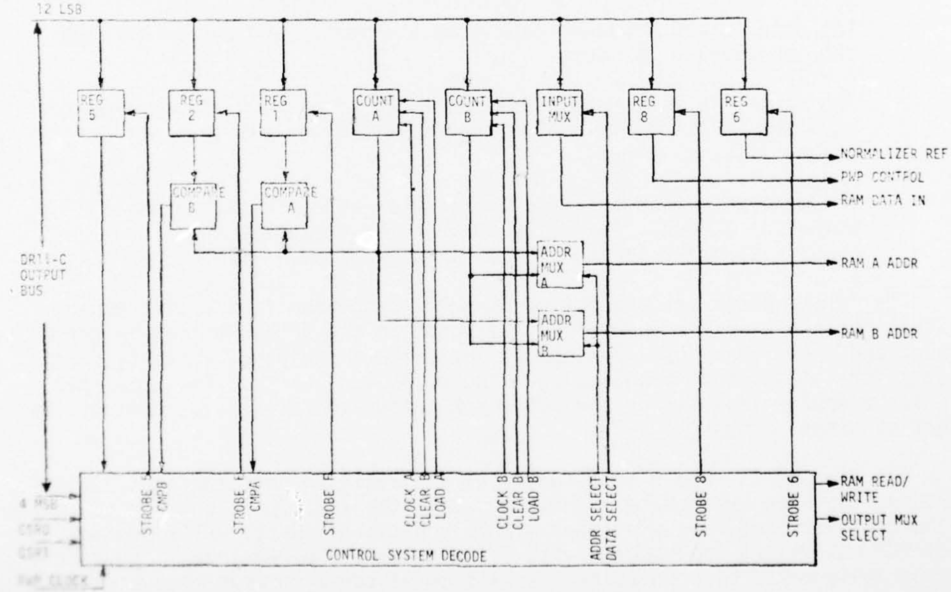


FIGURE 108. PWP TEST BED CONTROL SYSTEM

word consists of the 4 bit bus control field of the DR11-C data output bus, the 12 bit contents of REG5 loaded from the data field of the DR11-C data output bus, and the CSR0, CSR1 program defined bits. Figure 109 shows the control word organization. The bus control field is used to direct the contents of the data field into one of the registers tied to the DR11-C data output bus. The function code contained in REG5 is used to give the control system instructions

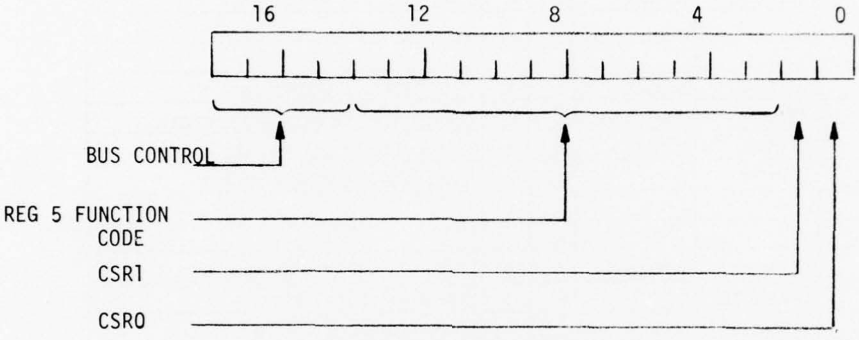


FIGURE 109. PWP TEST BED CONTROL WORD

about what type function it is to perform. CSR0 is used as a load pulse for latching data and CSR1 is used as a clock pulse for incrementing the address counter when the test bed is in the load mode.

Data or instructions are loaded into the control system by placing the data to be transmitted in the data field of the DR11-C output word and the destination in the bus control field. The test bed control system decodes the bus control field and activates the destination register, counter or memory. When CSR0 is pulsed, the data field is loaded into the destination.

The first instruction of any operation is to STOP and clear the system. Then REG5 is loaded with the function code for the operation. The control system decodes REG5 and primes the system for the operation. If the operation is a test, then REG1, REG2, REG6, REG8 and the address counters are loaded with test parameters. If the operation is a data load or data read operation, the address counters are preset to the starting address and data is then directed to the memories by decoding REG5 and the bus control field. After the memory has been completely loaded and test parameters set, the START command is given and the control system assumes control of the test bed. Control is returned to the computer only when the test is complete and the computer is flagged by REQA. Table 34 shows the instruction set for a typical operation. The memory is loaded with data, the test parameters are set and the START instruction is given. The program shown is for a 44 bit subsystem test operation.

A block diagram for the high speed memory and multiplexer is shown in Figure 110. The memory is divided into 11 bit quadrants. The quadrants are paired into the base half and displaced half. This is to provide a unique address for each 11 bit word cell in the memory. Since the memories are only

TABLE 34
PWP TEST BED TYPICAL CONTROL PROGRAM

PROG STEP	BUS CONTROL FIELD	DATA FIELD	DESTINATION	CSRO	CSR1	FUNCTION
1	04	0000	CONTROL	X		STOP - INITIALIZE SYSTEM
2	05	0063	REG5	X		SET CONTROLS FOR LOAD OPERATION
3	17	0000	ADDR CNTR A	X		PRESET COUNTER FOR START ADDRESS
4	00	DATA 1A	MEMORY	X	X	LOAD DATA - A MEMORY ADDR 0
5	00	DATA 2A	MEMORY	X	X	LOAD DATA - A MEMORY ADDR 1
.		.				.
:		:				:
N+3	00	DATA NA	MEMORY	X	X	LOAD DATA - A MEMORY ADDR N
N+4	17	2000	ADDR CNTR A	X		CHANGE ADDR TO DISPLACED MEMORY
N+5	00	DATA 1B	MEMORY	X	X	LOAD DATA - B MEMORY ADDR 0
N+6	00	DATA 2B	MEMORY	X	X	LOAD DATA - B MEMORY ADDR 1
.		.				.
:		:				:
2N+4	00	DATA NB	MEMORY	X	X	LOAD DATA - B MEMORY ADDR N
2N+5	01	PARAMETER	CMPR A	X		LOAD COMPARATOR REG A
2N+6	02	PARAMETER	CMPR B	X		LOAD COMPARATOR REG B
2N+7	05	PARAMETER	REG5	X		SET CONTROLS FOR TEST
2N+8	08	PARAMETER	REG8	X		SET PWP CONTROL SYSTEM
2N+9	17	0000	ADDR CNTR A	X		PRESET ADDRESS COUNTER A
2N+10	16	0000	ADDR CNTR B	X		PRESET ADDRESS COUNTER B
2N+11	03	0000	CONTROL	X		START TEST

1K long, the address extension to 4K is in the control system which enables the read/write and multiplexer controls.

The multiplexers direct data into and out of the memories. The input multiplexers are controlled by the contents of REG5. They are generally static multiplexers which are fixed at the start of the test and not changed throughout the test. The output multiplexers are dynamic in that they are controlled by the address. Thus, the memories all receive the same address and the multiplexers provide the extension. The data inputs or outputs are either 11, 22, or 44 bits depending on the multiplexer setting.

The return data from the PWP in some cases must be normalized. This is accomplished in the normalizer circuit shown in Figure 111. The reference exponent is stored as a test parameter in REG6 at the start of the test. The data stream exponent is inverted and added to the reference to produce a difference. The reference is constrained to be at least as great as the maximum exponent encountered in the data stream to prevent a negative output from the adder. The difference word then shifts the magnitude towards the least significant digit in the scaler. A shift enable control is provided to disable the normalizer and allow the output to be unnormalized.

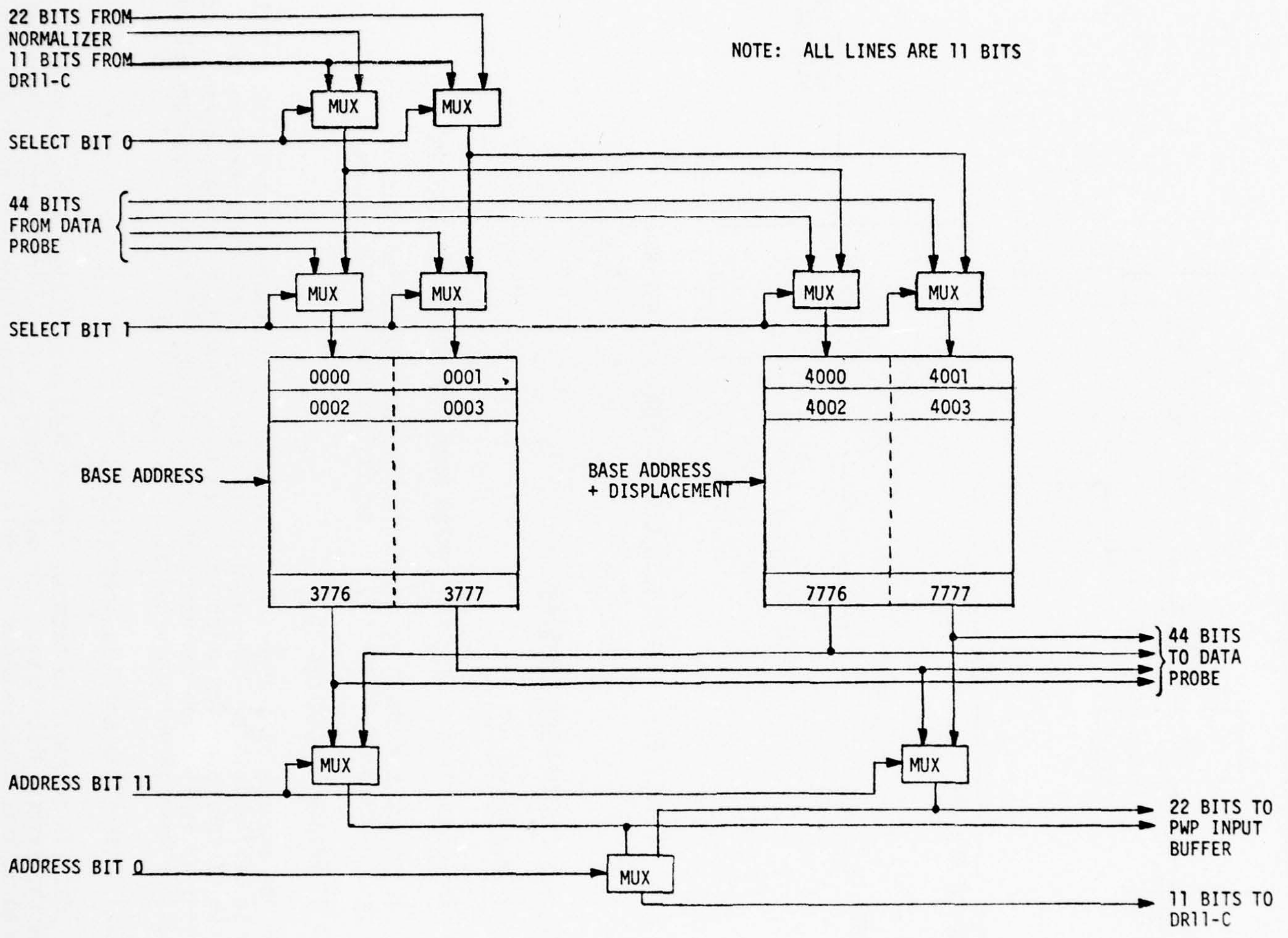


FIGURE 110. TEST BED MEMORY AND I/O MULTIPLEXING

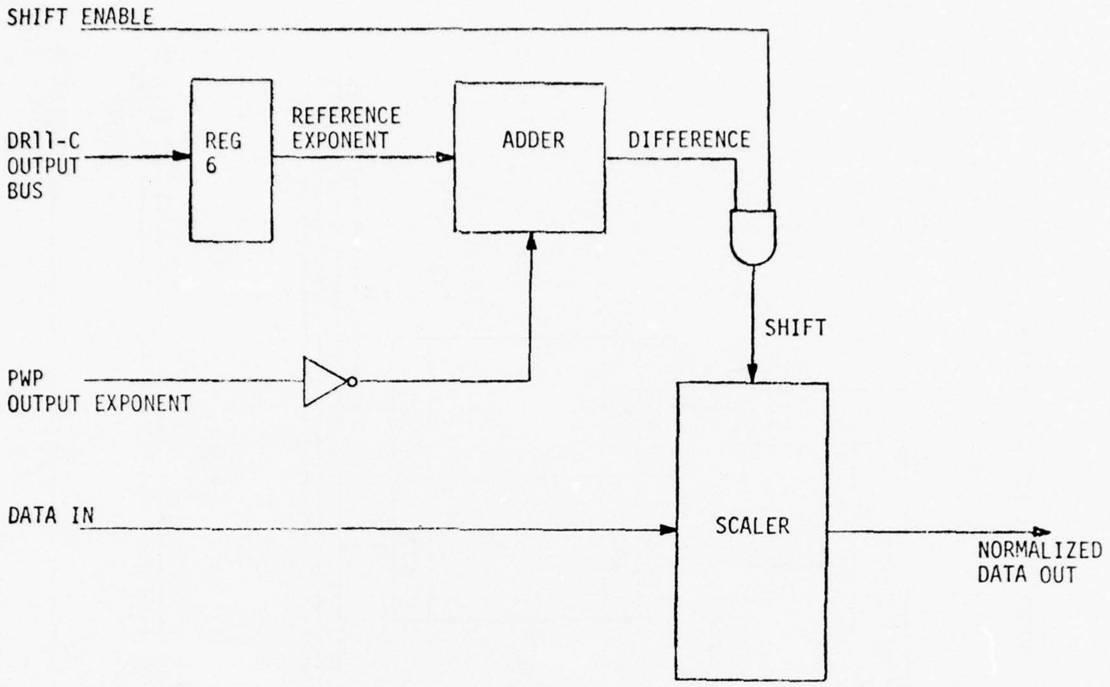


FIGURE 111. NORMALIZER CIRCUIT FOR PWP TEST BED

10.3 PWP SYSTEM TEST SOFTWARE

10.3.1 Objective

The objectives of the PWP system test program are to provide:

1. A trouble shooting aid during system integration, and
2. A means of exercising all or portions of the PWP system.

10.3.2 Program Organization

The program is a user interactive overlay structured system with a high degree of flexibility. The user directs the core resident program to call up one of four overlays which represent major functions in the program. Once an overlay is in core, the user can direct it to perform numerous specialized functions most of which are contained in subroutines.

10.3.3 Disk Files and Data Transmittal

Data is transmitted between overlays via disk files. A separate disk from the system disk is used for data storage. A listing of the various files stored on this disk is shown in Table 35. Source files such as file 1 and file 8 are read only files. They are serviced by separate programs so that there is no chance of unintentionally modifying them. Other files such as

TABLE 35
PWP SYSTEM TEST DISK FILES LOCATED ON DISK #1

FILE NUMBER	RECORDS/ FILE	WORDS/ RECORD	FUNCTION
1	15	80	SIMULATOR STEERING ARRAY
2	97	198	INPUT WAVEFORM LIBRARY
3	1	32	PROGRAM OPERATING PARAMETERS
4	6	32	SIMULATOR WORK SPACE
7	6	32	TEST BED WORK SPACE
8	30	64	SIMULATOR REFERENCE ARRAYS

the two work spaces and the parameter array are modified many times during system operation. File 2, the input waveform library is modified only occasionally when the user wishes to store a new waveform or edit an existing waveform. The total amount of contiguous disk space needed to store all files is 356 blocks.

Data transfers between subroutines within an overlay is accomplished through the use of common blocks and subroutine input output fields.

10.3.4 Overlay Functions

Figure 112 is a flow diagram of the major overlay functions for the PWP test program along with the disk files accessed by each overlay. The core resident program is called PWPTST and calls into core one of the overlays stored on the system disk. The four overlay functions are:

1. ARRAY - Input Waveform Manager
2. PWPSIM - System Simulator
3. TBIO - Test Bed Interface
4. PWPER - Error Checker

Figure 113 shows the subroutine stacking structure for each of the overlays.

10.3.4.1 Overlay ARRAY - This program is used to generate the input waveform library for the simulator program and the test bed hardware. The user has the following options at his disposal to perform this task:

1. Zero the library - This is the default option and requires user verification.
2. Input a new waveform.
3. Read a waveform from disk.
4. Edit an existing waveform.
5. Write a waveform to disk.
6. Print the directory for the waveform library.
7. Print a selected waveform.

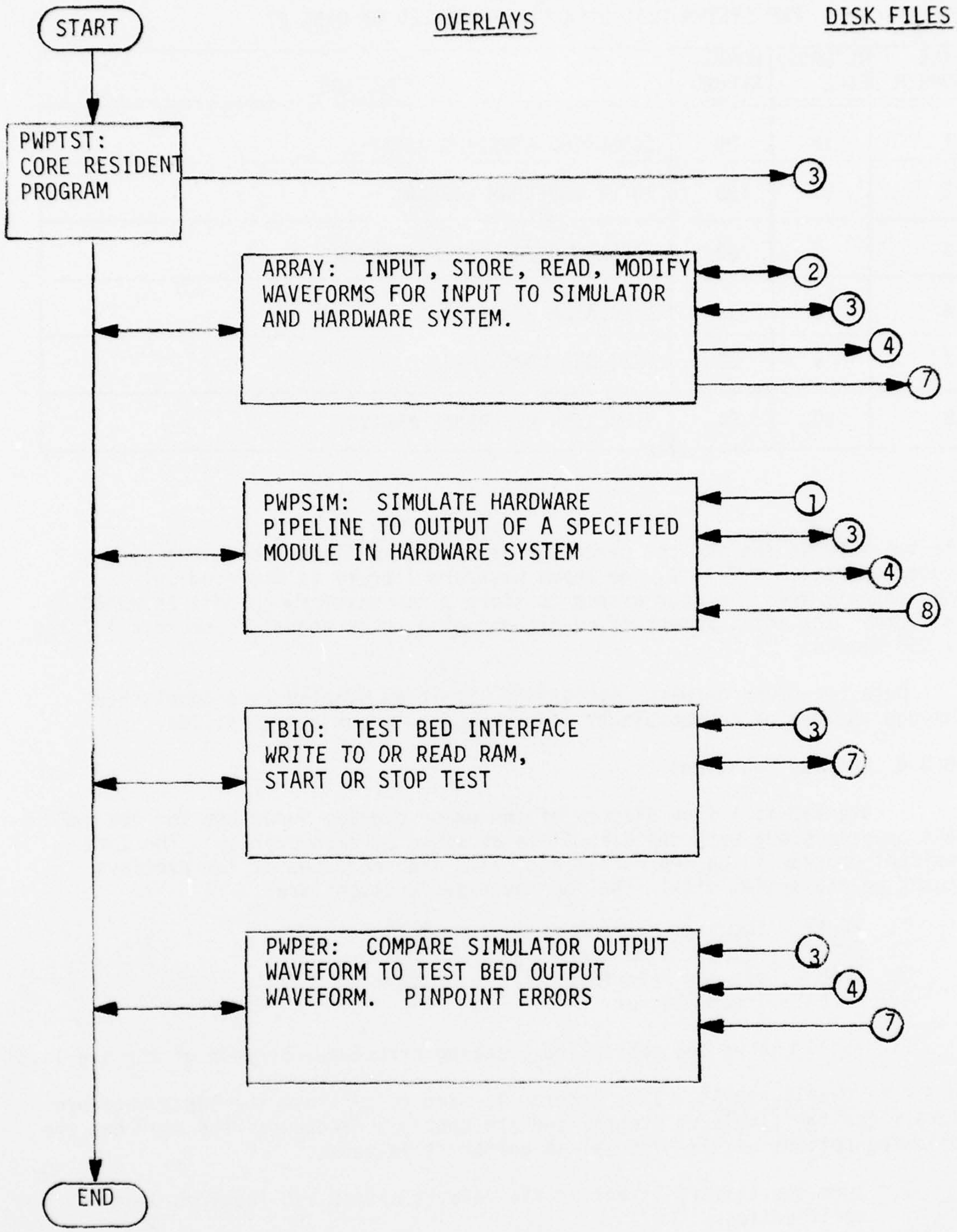

START
OVERLAYS
DISK FILES
PWPTST:
CORE RESIDENT
PROGRAM
3
ARRAY: INPUT, STORE, READ, MODIFY
WAVEFORMS FOR INPUT TO SIMULATOR
AND HARDWARE SYSTEM.
2
3
4
7
PWPSIM: SIMULATE HARDWARE
PIPELINE TO OUTPUT OF A SPECIFIED
MODULE IN HARDWARE SYSTEM
1
3
4
8
TBIO: TEST BED INTERFACE
WRITE TO OR READ RAM,
START OR STOP TEST
3
7
PWPER: COMPARE SIMULATOR OUTPUT
WAVEFORM TO TEST BED OUTPUT
WAVEFORM. PINPOINT ERRORS
3
4
7
END

FIGURE 112. OVERLAY FUNCTIONS AND DISK FILES ACCESSED FOR PWP SYSTEM TEST SOFTWARE

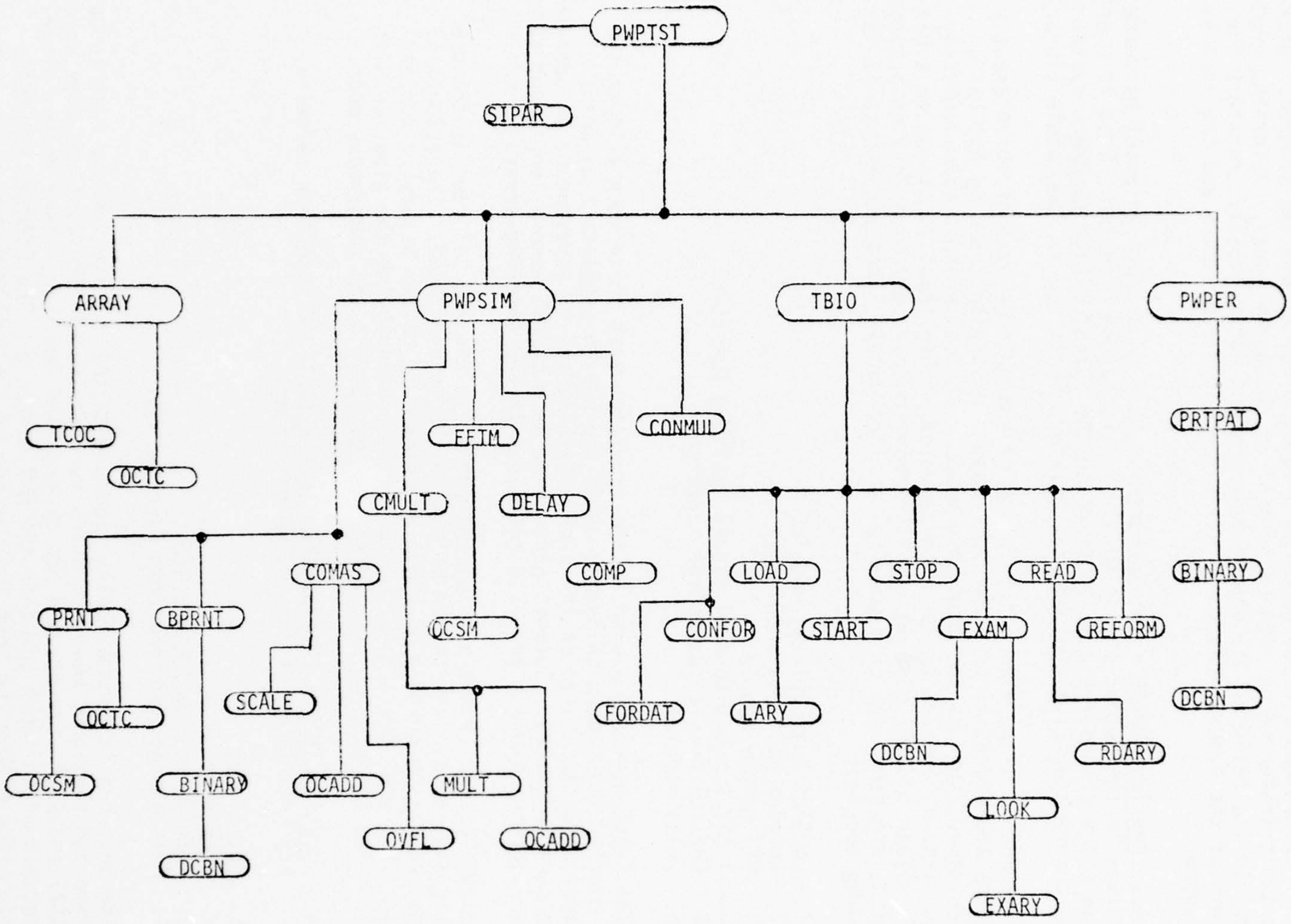


FIGURE 113. PWP SYSTEM TEST PROGRAM SUBROUTINE STRUCTURE

All operations are performed on a dummy array within the program. Disk files are transferred on request to or from the dummy array. Returning program control to the resident through an exit command automatically transfers the contents of the dummy array to both the simulator work space and the test bed work space.

10.3.4.2 Overlay PWPSIM - This program is used to perform an exact hardware simulation to the output of any module in the hardware system. This is a unique program which allows the user to simulate the output of any hardware system which can be constructed with the module simulators contained in the module library.

The module library is a collection of subroutines each of which exactly simulates the operation of a hardware module mostly on a bit by bit basis. In some cases, such as the complex add/subtract module and complex multiply module, simulations are broken down to perform chip level functions on a bit basis. Other modules such as the FFT memory module, which perform no arithmetic function, are simulated on a word basis. The module library consists of the following modules:

1. Complex add/subtract module.
2. Complex multiply module.
3. FFT memory module.
4. 1 bit x 8 bit control module multiply function.
5. Control module complement function.
6. Delay module.

The input waveform stored in the simulator work space upon exiting the ARRAY overlay is steered through each of the module simulators by means of the steering array stored on the disk. Each module is accessed by an address in the steering array. Support parameters such as FFT memory length and sine-cosine reference address are also contained in the steering array.

Simulator and hardware input and output points are selected by the user allowing any serial combination of modules to be simulated. The outputs of each module can be printed on the line printer in either binary or decimal format. This provides a powerful tool in troubleshooting the pipeline with a logic state analyzer while running the test bed in the continuous mode.

10.3.4.3 Overlay TBIO - This program is the test bed interface software. The user may select one of the following options:

1. Start test
2. Stop test
3. Examine test bed RAM content
4. Read output waveform from test bed RAM

The test bed is automatically loaded with the op-code for the test to be performed and the input waveform which is contained in the test bed work space. The START and STOP commands are non-destructive commands. They do not alter the contents of the work space or the RAM contents. The EXAMINE AND READ commands alter either the test bed op-code or the test bed work space. The only ways the RAM contents are altered are through a LOAD or a specific test. On the completion of a test, the RAM is read and upon exiting the program the RAM content is transferred to the test bed work space.

10.3.4.4 Overlay PWPER - This program reads the contents of the simulator work space and test bed work space and performs a bit by bit comparison of the information. The total number of errors are summed over the waveform and indicated on the video terminal. The user may elect to print out the waveforms in which case the input waveform, simulator output waveform and the test bed output waveform with error indicators are printed in binary format.

10.3.4.5 Support Programs - There are two external support programs for the PWP test program.

1. SETROM - Writes reference files to disk
2. SETSTR - Writes steering arrays to disk

The first program allows the user to input in decimal format those values which would be contained in the reference PROM's in the hardware. The user may select the file length and address for use by the steering array.

The second program allows the user to generate steering arrays which are used by the simulator. In this program, the user structures the hardware system to be simulated. Any system can be designed using the module building blocks contained in the module library.

This program architecture can easily be extended to a general purpose digital test system by making word quantization programmable and expanding the module library.

Preceding Page Blank - NOT FILMED

SECTION XI

MODULE TEST FACILITY

The module test facility, Figure 114 , was designed to provide complete test capabilities for all the PWG modules except the Universal TTL module. It allows faulty chips and/or wiring problems on the module to be detected and isolated so that the module can be quickly and accurately repaired. The system is designed to completely test the modules and to some degree isolate the problem to a chip or functional block without the use of internal probes. However, some probing is necessary to further isolate some of the more difficult faults such as cascaded faults.

It had previously been thought that faults could be isolated to a single chip input strictly by sequentially inputting data and controls and eliminating good data paths. The magnitude of this problem was not fully appreciated at the time and it was subsequently realized that we would not be able to develop this software within the scope of this program. An alternative approach was taken in which as many major signal paths as possible would be activated and the module response compared to a simulation of the module. In this way, coupled with a knowledge of the circuit, the operator could isolate the problem to a chip or logic area. Then by probing the circuit internally with probes, the problem is isolated. This procedure proved to be moderately fast. The average time to isolate a problem on a module is about 20 minutes.

The facility acts as a high speed interface between the PDP-11 computer and the module being tested. The computer generates input data, transfers it to the test facility and receives the module response from the facility. It then tests the module response data for correctness and informs the operator of any discrepancies. If the operator can determine the cause of the fault from this information, the test is complete. If not, he may select any input pattern he wishes for cycling through the test facility to enable him to probe the module and isolate the problem.

The test interface is divided into three sections. The input and storage section transfers data from the computer memory to the interface storage. Several data transfer cycles are required for this process. When all the data has been transferred, the input stage signals the second stage that it has completed the transfer. The second stage is the high speed sequencing and module timing stage. This stage sequences the stored data into the hybrid and produces the clock pulses required to clock the data to the module output. When the data has been acted upon by the module and is ready at the module output, the sequencing stage then signals the third stage that it is finished. The third stage, the output stage, then sequences the data back into the computer for examination.

The system is designed using TTL and TTL Schottky logic and consists of 150 circuits. It is mounted on two Cambion wire wrapped boards and is installed in a Cambion cabinet. The modules plug into sockets mounted on a plane at the top of the cabinet.

11.1 MODULE TEST HARDWARE

The module test hardware consists of a 62 bit parallel data register, data

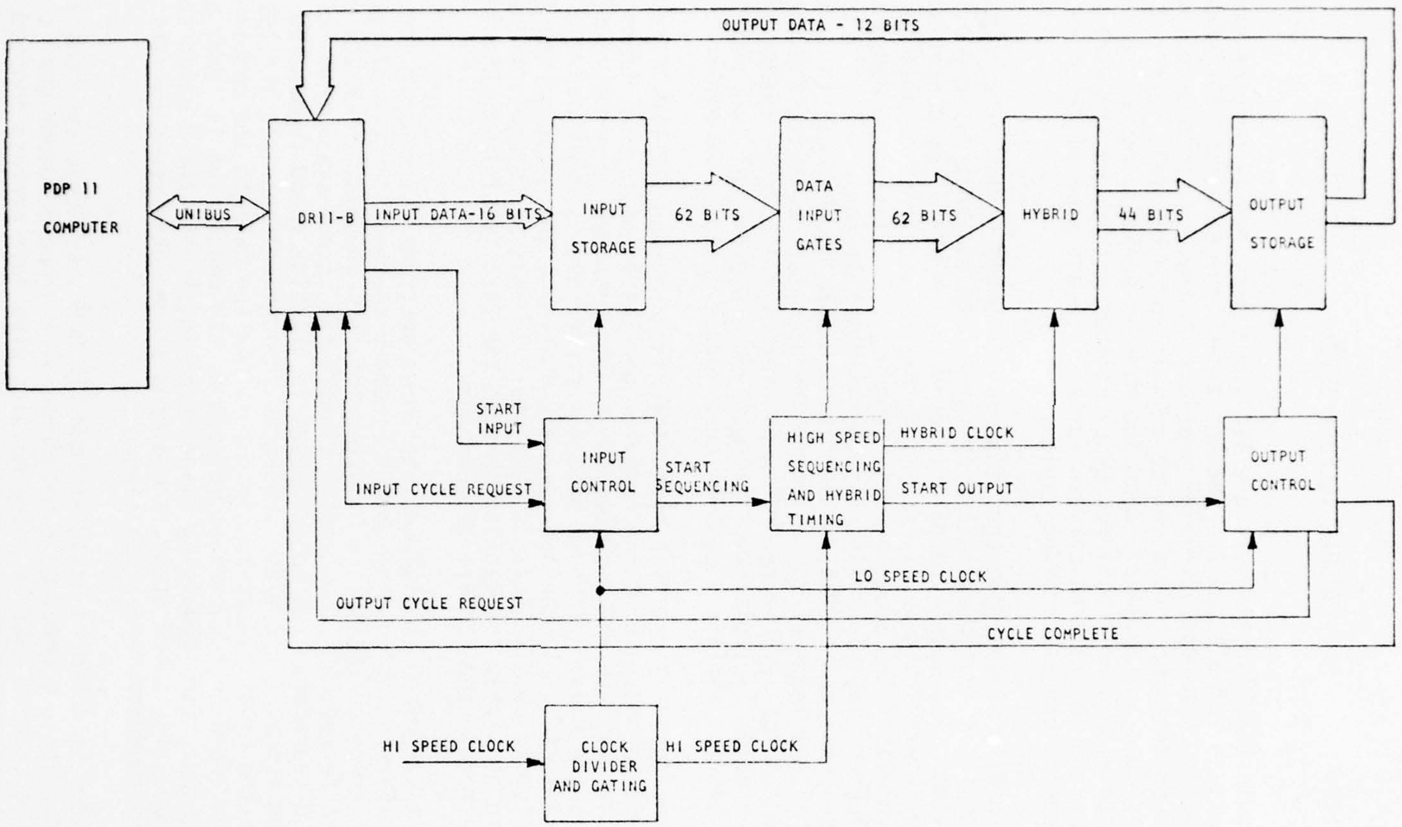


FIGURE 114 . MODULE TEST FACILITY

input gates, a 44 bit parallel data output register, and associated line drivers, level shifters and controls. The input and output registers are contained on one 9" x 9" Cambion wire wrap board. The control circuits and oscillator are contained on one 5" x 9" Cambion wire wrap board. The level shifters are in a separate chassis which houses the connector panel for the different modules. There is a separate connector for each module.

The tester operates by first loading the data input registers with the appropriate data and control bits from the computer. This operation requires 6 data transfer cycles since 86 bits must be loaded and only 16 lines are available on the DR11-C output bus. The 86 bits represent 62 bits of data and four 6 bit control words. The 62 data bits are formatted in different size input and control words depending on the type module under test. The four control words tell the test bed when to start the test, how to sequence the data into the module, and when to sample the module output.

Upon completion of the load cycle, the sequencing logic is given a signal to start testing the circuit. Data is then made available to the module under test in a manner dictated by the control words. The data sequencing circuit operates at the same speed as the module and is variable from 500 KHz to 10 MHz. When data should appear at the output of the module, a signal is given to the data output register at which time the states of the module outputs are latched. At the same time, a signal is given to the DR11-C requesting a data transfer from the test bed to the computer. This transfer requires 4 cycles because the output register is 44 bits wide and the DR11-C input bus is only 16 bits. After the module response is recorded by the computer, the test cycle is repeated with different data in accordance with the test program being executed by the computer.

11.1.1 Data Input Control

The data input control loads data into the test bed at the start of each test cycle. Figure 115 is a simplified logic diagram for this circuit. All operations are completely under the control of the computer.

The test cycle is initiated by the computer by strobing the CSR1 output of the DR11-C. This action clears all counters and resets all flip-flops in the test bed. The computer then places the first of the 6 input words on the DR11-C output bus. A 3 line to 8 line decoder in the test bed interprets the initialized state of the control counter and enables the latch gate for the word 1 data register. The computer then strobes the CSR0 output of the DR11-C. This action latches the contents of the output bus into the word 1 register. The CSR0 line is again strobed by the computer. This advances the control counter to the next input state. The decoder, in turn, enables the latch gate for the word 2 register and disables the latch gate for the word 1 register. The computer then places word 2 on the DR11-C data bus and the cycle is repeated. This procedure is repeated for all 6 input words. After the sixth word is loaded, the control counter is advanced to the seventh state. The seventh output of the decoder is activated which triggers an impulse from the one shot circuit. This impulse starts the data sequence circuit.

11.1.2 Data Sequencing Control

The data sequencing circuit controls when data is presented to the input

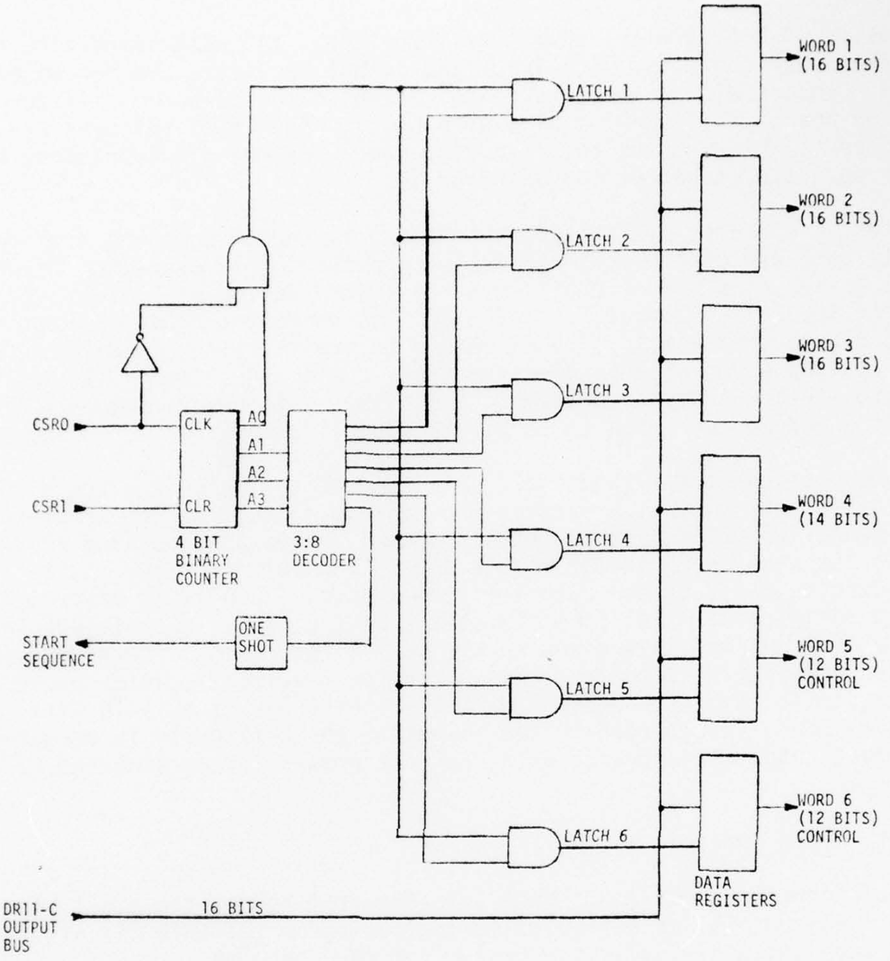


FIGURE 115. DATA INPUT CONTROLLER FOR MODULE TESTER

of the module under test. Figure 116 is a block diagram of the data sequencing control.

The START SEQUENCE line from the data input control initiates the sequencing circuit. When the START SEQUENCE line goes low, the control R-S flip flop is set allowing the high speed clock to drive the sequencing counter. Also, the dual flip flop which controls the 4:1 multiplexer is in the zero state as a result of the CSR1 strobe at the beginning of the test cycle. The 4:1 multiplexer allows the contents of the number 1 control register to be present at the comparator input. When the counter reaches that value indicated by the first control register, the sequencing cycle is started.

AD-A043 302   RCA GOVERNMENT SYSTEMS DIV MOORESTOWN N J MISSILE AND--ETC   F/G 17/9
PROGRAMMABLE FFT LINEAR FM WAVEFORM PROCESSOR. PHASES II AND II--ETC(U)
JUL 77   L W MARTINSON, J A LUNSFORD   F33615-74-C-1077

UNCLASSIFIED   AFAL-TR-77-23   NL

3 OF 3
AD
A043302

END
DATE
FILMED
9-77
DDC

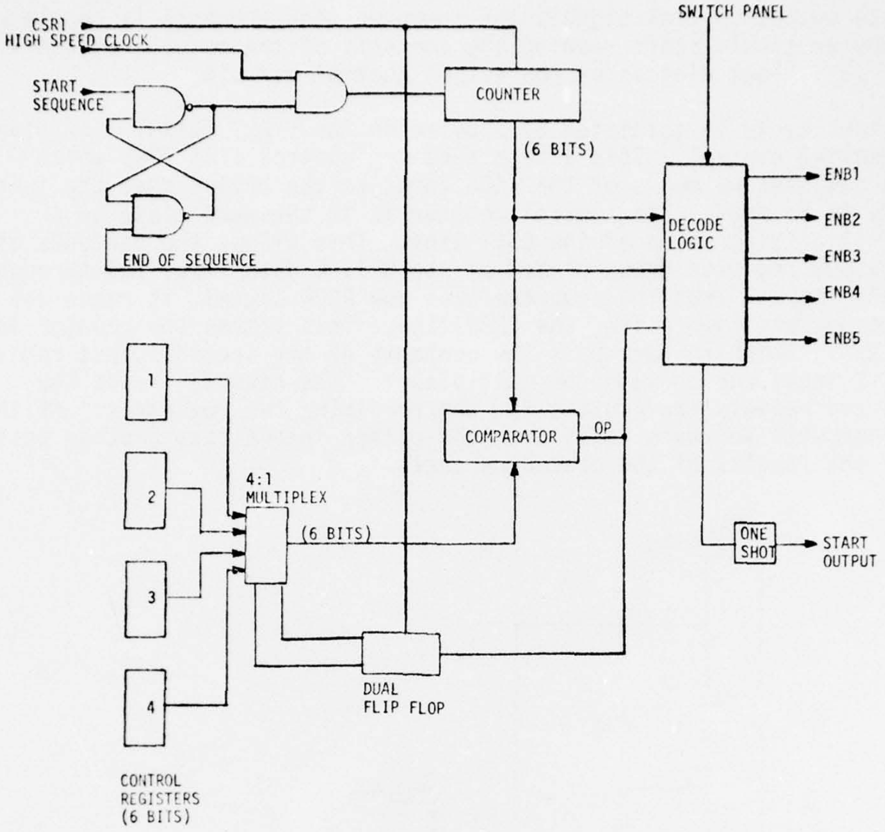


FIGURE 116. DATA SEQUENCING CONTROL FOR MODULE TESTER

The decode logic responds to the switch panel setting, the counter output and the OP signal from the comparator to enable the data input gates by activating the ENB1 to ENB5 control lines. Which of the 5 lines are enabled depends on the type module being tested.

When the first comparison occurs, the OP signal also advances the dual flip flop to its second state. This allows the contents of the second control register to be placed at the input of the comparator. This word either represents the test termination point or the point at which another input enable gate is activated. Which of these functions is valid is determined by the switch panel. The comparison cycle continues until an end of test word is experienced. At this point, the one shot is strobed causing the output state of the module under test to be latched in the output register and the output control circuit to be activated.

11.1.3 Data Output Control

The data output control signals the computer that the test is complete and the computer should start reading the contents of the output registers. Figure 117 is a block diagram of the output control circuit.

The output cycle is initiated by a pulse on the START OUTPUT line given by the sequencing control. This action sets the control flip flop which signals the computer by means of the REQA input to the DR11-C that the test bed is ready to be read. The control counter is in the zero state as a result of initializing pulse of the CSR1 line. This allows the contents of the first output register to be placed on the DR11-C data input bus through the 4:1 multiplexer. When the computer sees the REQA signal, it reads the DR11-C input bus and then pulses the CSR0 line. This places the counter in the second state which in turn puts the contents of the second output register on the DR11-C input bus through the multiplexer. The computer reads the second word and repeats the process for the remaining two registers. At this point, the computer software takes over and either initializes another test or analyzes the results of the preceding tests.

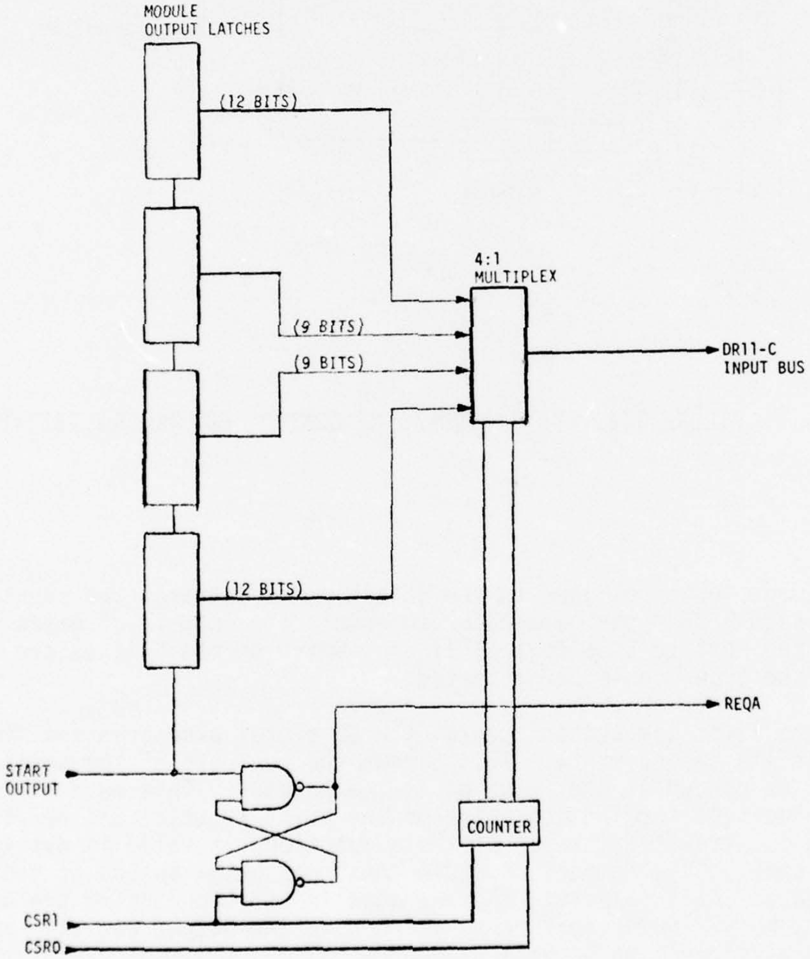


FIGURE 117. DATA OUTPUT CONTROL FOR MODULE TESTER

11.2 MODULE TEST SOFTWARE

Test programs have been written for 6 of the PWG modules. These modules are:

1. FFT Memory
2. Complex Multiplier
3. Adder/Subtractor
4. Control
5. Level Translator
6. Universal SOS - Retimer Function

No software was written for the reorder memory module because this module was not implemented in the system. Also, the universal TTL module software was omitted because all functions implemented with these modules are unique and the amount of software required would be enormous.

Different approaches were taken with each test program based on module complexity, program size, data storage requirements, program run time, and experience gained from early programs. Consequently, early programs are large and slow, while latter programs are more compact and fast. Learning how to circumvent deficiencies in the PDP-11 system software was a large factor in this.

11.2.1 Test Program Philosphy

Most of the modules in the PWP incorporate a high degree of large scale integration and all modules have a large number of inputs. The complex multiplier module, for example, has a total of 10,500 devices and 36 inputs. To simply test all possible combinations of inputs would require 2^{36} iterations. The storage requirements and program run time would make this approach too impractical. To check all possible stuck-at-zero or stuck-at-one conditions for all devices is clearly an unattainable goal.

For these reasons, a functional approach was chosen. In this approach, the different LSI chips used to build the different modules were carefully studied and a set of input patterns chosen for that chip which would exercise most or all major signal paths. An example of this is the type A full adder in the TCS-065 adder chip.

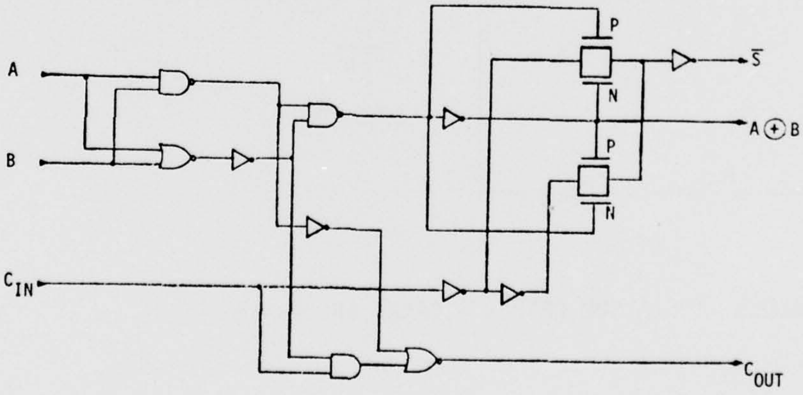


FIGURE 118. TYPE A ADDER CIRCUIT ON TCS-065 ADDER CHIP

Figure 118 shows the type A adder logic diagram. The circuit inputs are A, B and C_{in}. The outputs are $\overline{S}$ A $\oplus$ B and $\overline{C}_{out}$. The inputs each must make at least one transition from low to high, and high to low in the test in such a manner as to stimulate at least one low to high, and high to low transition on each output. The patterns must exercise as many gates in the circuit as possible. In order to do this, the circuit is broken down into different input to output signal paths. Figures 119 to 122 show four such major signal paths for the type A adder. After the various combinations for the other circuit types in the adder chip are taken into account, a set of input test patterns is compiled which exercises most of these paths. Table 36 represents a typical test pattern set for the TCS-065 adder chip.

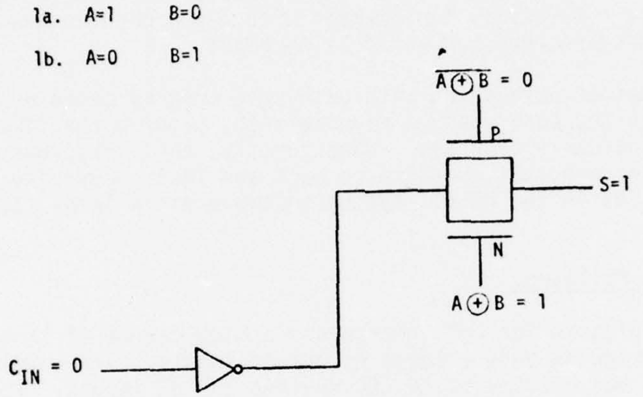


FIGURE 119. ADDER TYPE A SUM PATH 1 - NO CARRY IN, A ≠ B

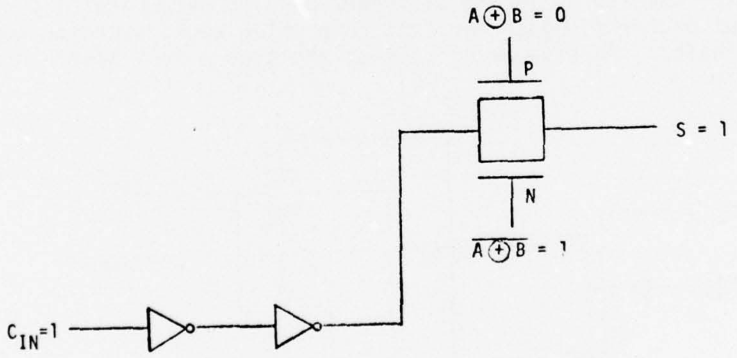


FIGURE 120. ADDER TYPE A SUM PATH 2 - CARRY IN, A = B

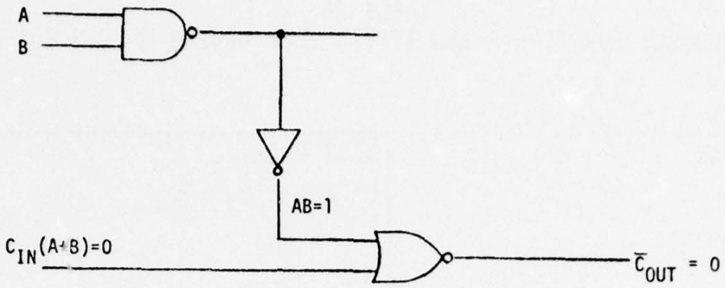


FIGURE 121. ADDER TYPE A CARRY PATH 1

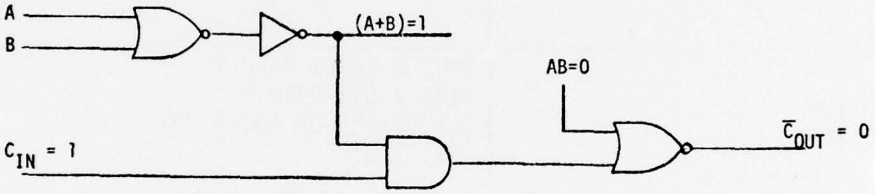


FIGURE 122. ADDER TYPE A CARRY PATH 2

A set of module input patterns is then designed to provide the chip level set of inputs to each chip on the module. In this way, every chip on a working module sees the full set of test patterns. This system breaks down of course if cascaded faults occur on a chip level. However, the software is designed to allow manual intervention and test pattern manipulation by the test technician to isolate these faults. The drawback is that the technician must be very familiar with the circuit operation.

TABLE 36
TEST PATTERN SET FOR TYPE A AND TYPE B FULL ADDERS ON TCS-065 ADDER CHIP

INPUT PATTERN A: A_7 - A_0 B: B_7 - B_0	TEST PATH OUTPUT: C_7 - C_0
1. A: 1111 1111 B: 0000 0001	TYPE A&B CARRY PATH 2 (EXCLUSIVE OF ADDER #0) OVERFLOW SHIFT C=1000 0000
2. A: 0101 0101 B: 0101 0101	TYPE A CARRY PATH 1 TYPE B SUM PATH 2 C = 1010 1010
3. A: 1010 1010 B: 1010 1010	TYPE B CARRY PATH 1 TYPE A SUM PATH 2 (EXCLUSIVE OF ADDER #0) C=0101 0101
4. A: 1111 1111 B: 0000 0000	TYPE A & B SUM PATH 1A C = 1111 1111
5. A: 0000 0000 B: 1111 1111	TYPE A & B SUM PATH 1B C = 1111 1111

The test bed programs, although different in structure, have the same basic macro functions. Each program has a test pattern generation program which either has all necessary patterns stored or allows the technician to input the patterns. Each program simulates the hardware response to the input pattern. Each program has an automatic test program which steps through all test patterns in the pattern library. Each program has a comparison program which matches the hardware response to the simulator response and indicates to the technician the number of erroneous responses. And finally, each program has a manual intervention program which allows the technician to cycle a selected pattern through the hardware and observe the response by means of an oscilloscope or logic state analyzer.

11.2.2 FFT Memory Test Software

This program is an overlay structured user interactive program with the following features:

1. Test Pattern Generator and Hardware Simulator
2. Automatic Module Tester
3. Manual Intervention
4. Individual Chip Test with Error Logger for Intermittent Problems
5. Various Error Mask and Output Modes

This program is the first test software written for the PWP and, although it performs its designed goals, it is inefficient and requires 10 to 15 minutes to run. There are two major reasons for this. First, the program is bit oriented and is written in FORTRAN language. We have learned that assembly language is far superior and faster for bit oriented operations. Second, all data transfers between overlays and within overlays are performed by disk transfers. The reasons for this are the error in the PDP-11 system software which prevents common block transfers between overlays, and the requirement by this program for very large data array due to the FORTRAN language. Subsequent programs were written in assembly language and avoided the overlay structure by reducing array and program size.

11.2.2.1 Disk Files and Data Transfers

All disk files with the exception of the chip test program's error logger are on the system disk. The error logger is on disk 1. Table 37 lists the disk files and their functions. The requirement for so many disk files is that the binary format in a FORTRAN program uses a whole word for each bit and thus requires huge arrays. Usable core space is limited and so these arrays must be stored on the disk. The total disk space required for this program is 2945 contiguous blocks. This is almost one quarter of the disks storage capacity.

All transfers between overlays are by disk transfer. Some subroutine transfers are by disk transfers while some are by common block.

11.2.2.2 Overlay Functions

Figure 123 is a flow diagram of the major overlay functions with the disk files accessed. The core resident program is FMT and calls one overlay at a

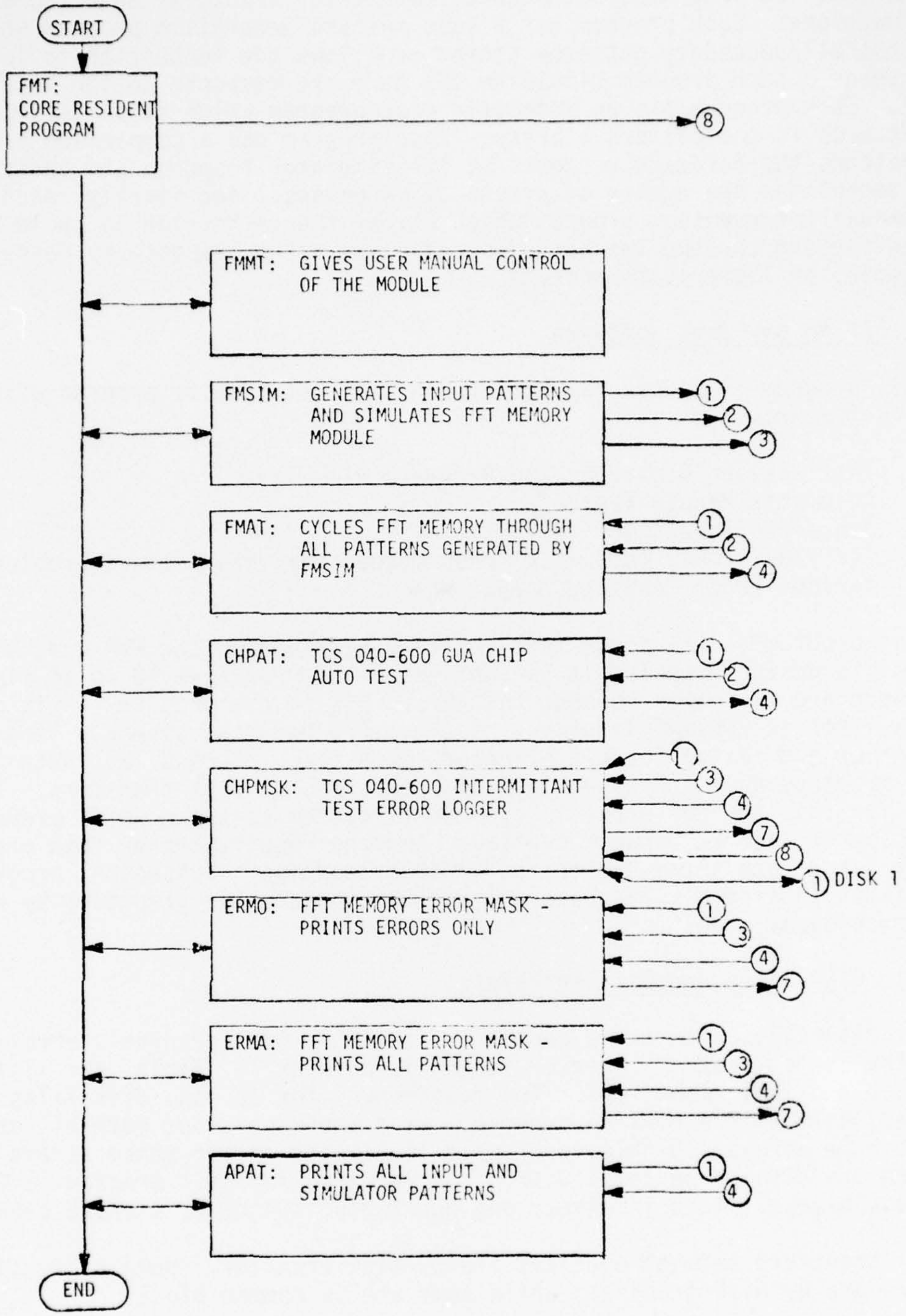

START
FMT:
CORE RESIDENT
PROGRAM
8
FMMT: GIVES USER MANUAL CONTROL
OF THE MODULE
FMSIM: GENERATES INPUT PATTERNS
AND SIMULATES FFT MEMORY
MODULE
1
2
3
FMAT: CYCLES FFT MEMORY THROUGH
ALL PATTERNS GENERATED BY
FMSIM
1
2
4
CHPAT: TCS 040-600 GUA CHIP
AUTO TEST
1
2
4
CHPMSK: TCS 040-600 INTERMITTANT
TEST ERROR LOGGER
1
3
4
7
8
1 DISK 1
ERMO: FFT MEMORY ERROR MASK -
PRINTS ERRORS ONLY
1
3
4
7
ERMA: FFT MEMORY ERROR MASK -
PRINTS ALL PATTERNS
1
3
4
7
APAT: PRINTS ALL INPUT AND
SIMULATOR PATTERNS
1
4
END

FIGURE 123. OVERLAY FUNCTIONS AND DISK FILES ACCESSED FOR FFT MEMORY MODULE TEST SOFTWARE

TABLE 37

DISK FILES FOR FFT MEMORY MODULE TEST PROGRAM, DISK 0 AND 1

FILE NUMBER	RECORDS/ FILE	WORDS/ RECORD	FUNCTION
1	2048	15	MODULE INPUT PATTERNS (BINARY FORMAT)
2	2048	6	DELAY PARAMETER
3	2048	6	SIMULATOR OUTPUT
4	2048	26	TEST BED OUTPUT (BINARY FORMAT)
7	2048	37	ERROR LOG (BINARY FORMAT)
8	1	42	PROGRAM PARAMETERS
1 DISK 1	2	2048	INTERMITTANT TEST ERROR LOG

time into core in response to directions given by the user. The overlay functions are:

1.	FMSIM	- Simulator
2.	FMAT	- Automatic Test
3.	FMMT	- Manual Intervention
4.	CHPAT	- Chip Auto Test
5.	CHPMSK	- Chip Test Error Mask
6.	ERMO	- Error Mask
7.	ERMA	- Error Mask
8.	APAT	- File Print

Figure 124 shows the subroutine structure for the program.

11.2.2.3 Overlay FMSIM

This program generates the test patterns for input to the simulator and module by generating all possible combinations of the 11 inputs to a GUA chip and simulates the FFT memory module. This method of pattern generation is used because the number of inputs are small. There are 2 data inputs for each of 11 chips on the module while there are 9 control inputs which are shared by all the chips. The 2 data inputs are not considered as unique for each chip because there is no interaction between chips. Therefore, there are only 9 control and 2 data inputs for 11 unique inputs. These represent 2048 unique input patterns.

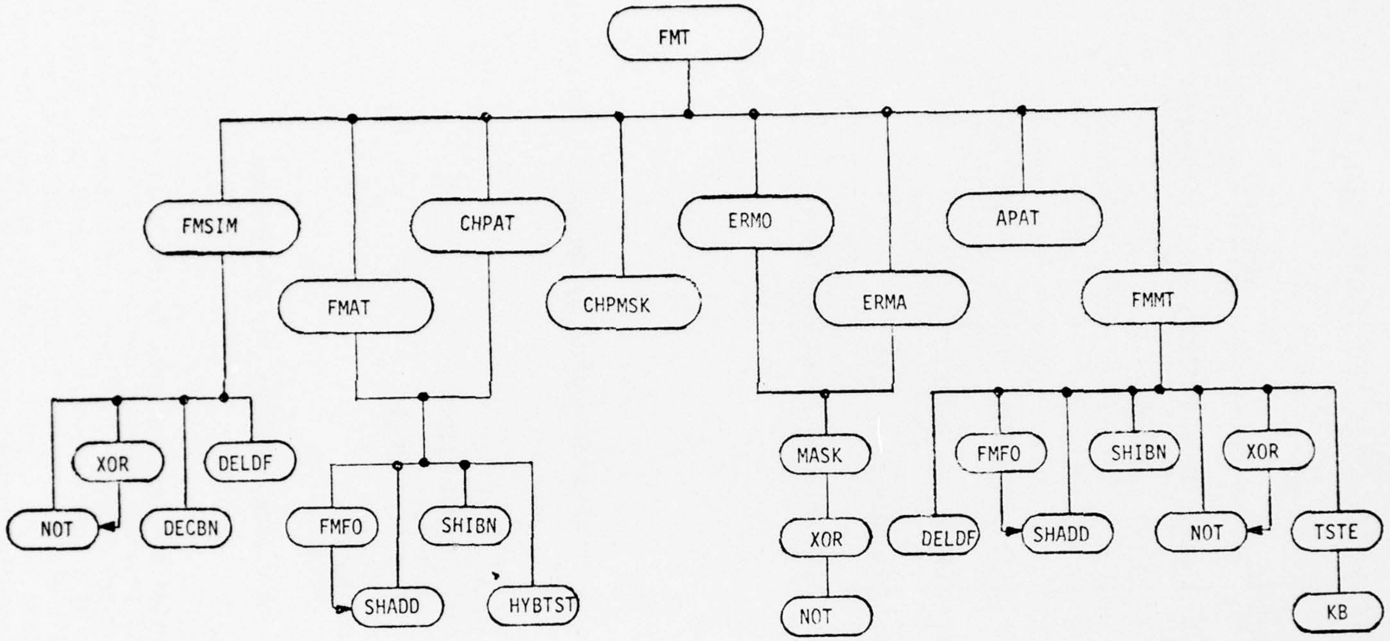


FIGURE 124. SUBROUTINE STRUCTURE FOR FFT MEMORY MODULE TEST PROGRAM

The FFT memory is simulated by a set of Boolean equations which model the memory functions. If the two data inputs are designated S and D and the outputs FM and FPM, then the outputs can be specified in terms of S and D and the control inputs IS, FS, OS, SF, SI, and CR. The equations are:

$$XA = S \cdot \overline{IS} + D \cdot IS$$
$$XB = D \cdot \overline{IS} + S \cdot IS$$

$$XC = XA \cdot \overline{FS} + XB \cdot FS$$
$$XD = XB \cdot \overline{FS} + XA \cdot FS$$
$$XE = XC \cdot \overline{OS} + XD \cdot OS$$

$$CXE = XE \oplus \overline{SI}$$
$$CXC = XC \oplus \overline{SF}$$

$$FM = \overline{CXC} \oplus CR$$
$$FPM = \overline{CXE} \oplus CR$$

The simulator must also calculate the time delay from input to output for a pulse to pass through the memory. This is accomplished by a look up table which uses the A, B, C length control bits as an address. The delay parameter is stored for each input pattern and later fed to the test bed to control the output sampling point.

11.2.2.4 Overlay FMAT

This program performs data formatting operations and input/output operations to the test bed for the total test pattern library. Prior to outputting data to the test bed, the data must be formatted to occupy the 16 data lines from the computer in such a way as to occupy the proper registers in the test bed to match the input requirements of the FFT memory module. The test parameters must also be formatted to occupy the proper data lines. After completion of the test and the output is latched in the test bed output buffer, the program reads the data back into the computer and reformats it for processing by the error analysis program.

11.2.2.5 Overlay FMMT

This program is the manual intervention program which allows the user to repeatedly cycle a given input pattern through the module under test. The user selects the input pattern which is then put through the hardware simulator before being sent out to the hardware. The pattern is then continuously cycled through the hardware until a signal is given by the user to terminate the test. Upon receipt of this signal, the test bed latches the module output and sends it back to the computer. The program then performs a comparison of the module response to the simulator response and displays the exclusive OR of these two patterns on a bit basis.

11.2.2.6 Overlays ERMO, ERMA, APAT

These three programs are test output programs. Two of them perform an error mask of the test bed output by performing an exclusive OR between each bit of the simulator response and the module response. This exclusive OR is stored along with the input pattern. In the ERMO overlay only the errors are stored and subsequently printed. In the ERMA overlay, all patterns, erroneous or not, are stored and subsequently printed. The APAT overlay does not perform an error mask and is used solely to print out the test pattern library and the simulator response library.

11.2.2.7 Overlays CHPAT and CHPMSK

These programs were added to the FFT memory test program after it was discovered that there was an instability in the TCS-040-600 chip. At the time there was no way to effectively test these devices other than to continuously cycle a test pattern through a chip and observe the response on an oscilloscope. This was very time consuming because there were no laws governing the time to failure of a device or which test pattern if any would cause the failure.

These programs were written to continuously cycle through the test pattern library, sample the chip response for each pattern, compare the output to the simulated response and log an error cycle if an error did occur. The program CHPAT performs the same basic functions as FMAT except that in this case only one bit is used instead of 11 required for the FFT memory module. The program CHPMSK takes the chip response and compares it to the simulator response library. If an error occurs, the number of erroneous patterns is recorded in the error logger disk file for that cycle. In this way, the time to failure is accurately recorded. A record of the amount of degradation with time is also obtained by observing the number of failures with time. If the chip should recover, this would be indicated by a failure recorded and then no failure recorded for the next cycle. Up to 2048 cycles can be recorded. The cycle time of the program is 7 minutes so 239 hours of continuous testing is allowed. It was found that in no case did a failure occur after 6 hours and no recoveries were observed.

11.2.3 Complex Multiplier Test Software

This program is a subroutine structured user interactive program with the following features:

1. Test Pattern Generation
2. Module Simulator
3. Automatic Test
4. Manual Intervention
5. Error Analysis

This program represents an intermediate step in the realization of a truly efficient program. The data arrays were reduced in size allowing a subroutine structure instead of the overlay structure. This was accomplished by using word oriented data. By going to a subroutine structure, all data transfers could be accomplished by means of common block transfers. This reduced program execution time by a factor of ten.

11.2.3.1 Disk Files and Data Transfers

Although data transfers between subroutines are accomplished through common blocks, disk files are not excluded from the program. They are still required for permanent storage of test patterns and simulator response. Also for convenience the test bed output and error log is stored on disk. Table 38 lists the disk files accessed by the complex multiplier test program. The total amount of contiguous disk space required is 1056 blocks.

TABLE 38
DISK FILES FOR COMPLEX MULTIPLIER MODULE TEST PROGRAM, DISK 0

FILE NUMBER	RECORDS/ FILE	WORDS/ RECORD	FUNCTION
1	1024	4	INPUT PATTERNS
2	1024	4	SIMULATOR OUTPUT
3	1024	4	TEST BED OUTPUT
4	1	16	PROGRAM PARAMETERS
7	1024	17	ERROR LOG

11.2..3.2 Major Subroutine Functions

Figure 125 shows the major subroutine functions and disk files accessed by each subroutine of the complex multiplier program. The main program is CMT and calls on the different subroutine functions in response to instructions by the user. The subroutine functions are:

1. CMPRM - Input Pattern Generator
2. CMSIM - Module Simulator
3. CMAT - Automatic Test
4. CMMT - Manual Intervention
5. CMEM - Error Analysis

Figure 126 shows the subroutine stacking structure for the program.

11.2.3.3 Subroutine CMPRM

This subroutine generates the test patterns for the multiplier module. It has the first 512 patterns fixed and allows the user to add patterns to the library or change a pre-recorded pattern. The program operates in the edit mode where the user moves a pointer through the data library and executes add or change instructions to modify the library. When the user exits the program, the library contents are permanently stored on disk for access by other subroutines. In this way, the pattern generator does not need to be executed every pass through the program.

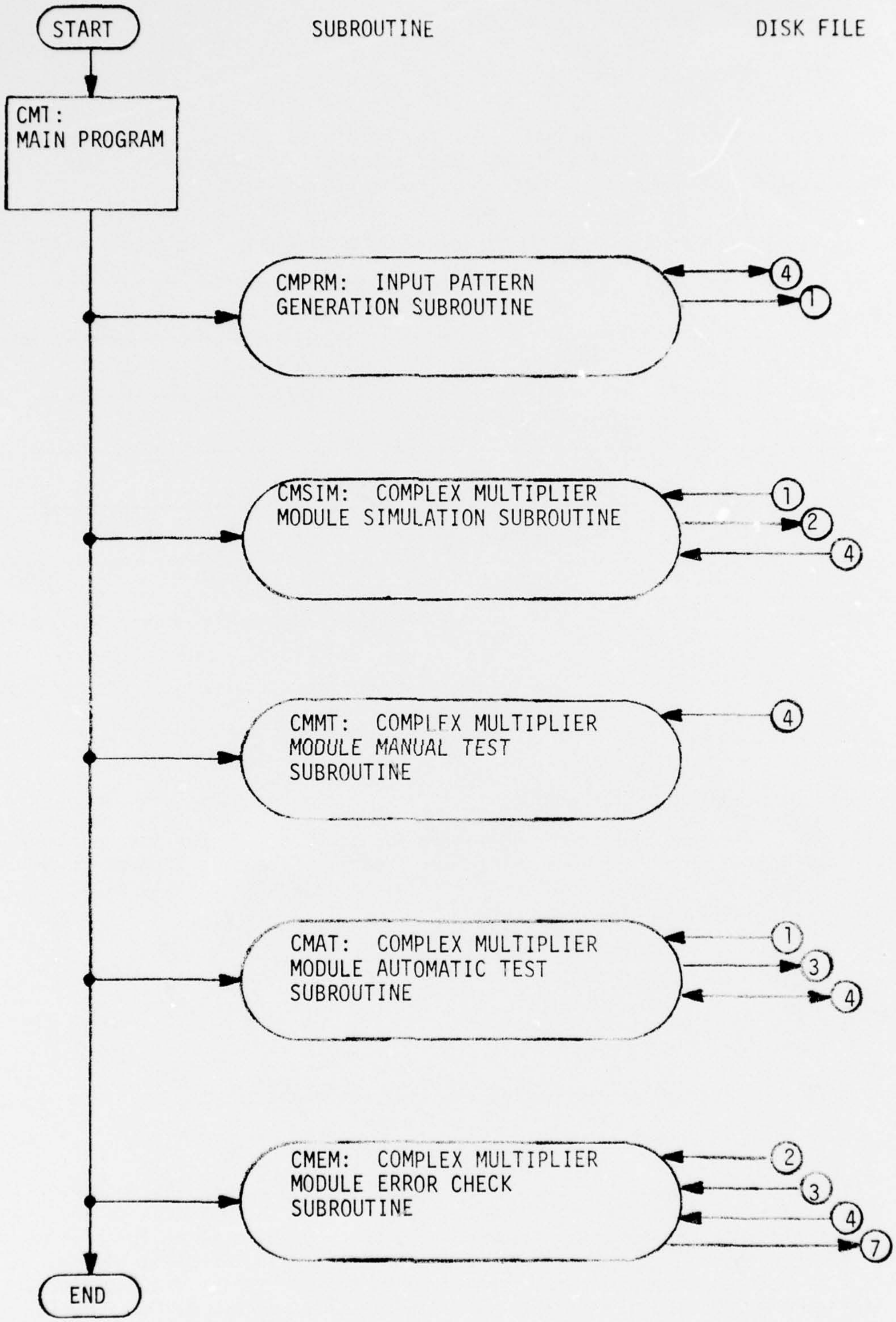


FIGURE 125 . MAJOR SUBROUTINE FUNCTIONS AND DISK FILES ACCESSED FOR COMPLEX MULTIPLIER MODULE TEST SOFTWARE

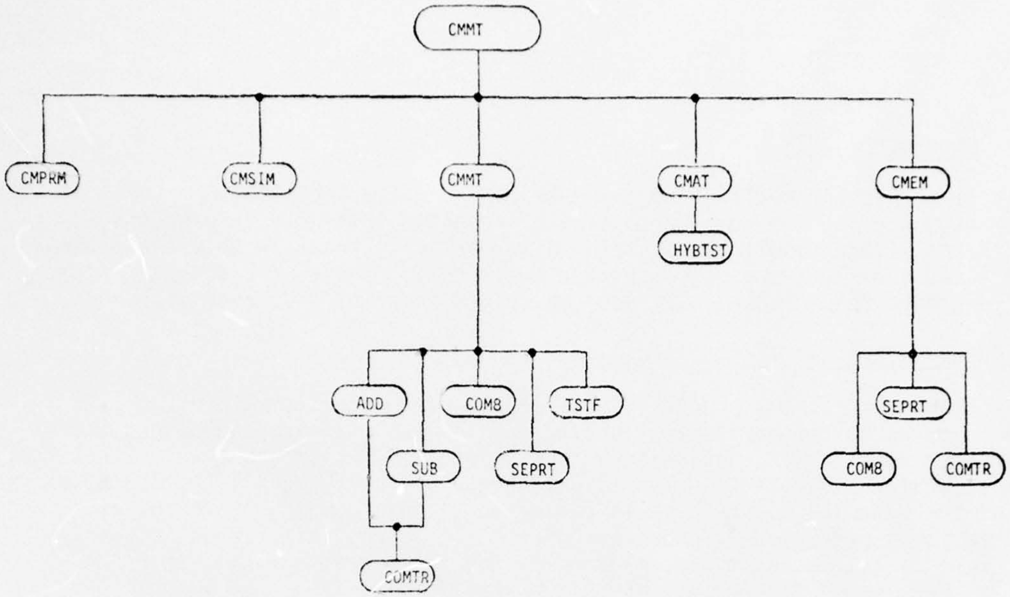


FIGURE 126. SUBROUTINE STRUCTURE FOR COMPLEX MULTIPLIER MODULE TEST PROGRAM

11.2.3.4 Subroutine CMSIM

This subroutine simulates the complex multiplier response for each of the patterns stored in the test pattern library. The program is written in FORTRAN language and executes an algorithm to simulate the multiplier response. This program does not exactly model the hardware on a bit basis but does perform the same function. Inputs and outputs are handled on a word basis and multiplication, addition, and subtraction utilize the FORTRAN library programs for these functions. The simulated responses to the input library are stored permanently on disk for the same reason as the data library is stored. Whenever a change is made to the data library, the simulator must be re-executed. Otherwise, the complex multiplier test program may be run over and over without executing either the test data generator or the simulator.

11.2.3.5 Subroutine CMAT

This subroutine handled the data input/output and formatting functions for the complex multiplier automatic test. Data is read from the library, formatted for the test bed buffer configuration required for the multiplier, and sequenced out to the module. The module response is read back in, reformatted and stored on disk for utilization by the error analysis program.

11.2.3.6 Subroutine CMMT

This subroutine is the manual intervention program. The user may select his own input to the module or read one from the data library. The pattern is then put through the simulator and all intermediate responses are printed. The pattern is then repeatedly circulated through the module to allow the user to observe the module response. Upon a signal from the user, the test is terminated and the module response is read back into the computer. The simulator response and the module response are then displayed to let the user observe the difference.

11.2.3.7 Subroutine CMEM

This is the comparator routine for determining differences between the simulator response and the complex multiplier response. The two responses are subtracted and if the result is not zero, an error is logged. In this case, the input pattern, the simulator response, and the module response are stored. After all patterns have been checked, the user may elect to print the error library.

11.2.4 Adder/Subtractor Test Software

This program is also a subroutine structured user interactive program with the same features as the complex multiplier module test software. However, there are two differences between this program and the multiplier program. The first is that the simulator is written in assembly language and performs all the operations exactly as the hardware does. Data is converted to ones complement format and all arithmetic is performed in ones complement. The second difference is that magnetic tape is used as a storage medium for input patterns and simulator responses. The reason for this is that it was anticipated that the system test data would be so bulky that magnetic tape would be the only convenient storage medium and we wanted to develop and test this concept on a smaller scale.

11.2.4.1 Disk Files and Data Transfers

Table 39 lists the disk files accessed. There are significantly fewer disk files in this program than in past programs. Most data is held in common blocks and the data in those blocks are modified by the program. Thus it is only necessary to read the input patterns from the disk once each time the program is run. The test patterns are also stored on tape so that on the first pass through the program the tape must be accessed and data transferred to the disk. As stated above, this method is to test the concept for the system test program. However, another reason is that there is no algorithm to generate the test pattern sequence and since the patterns must be input manually, some non-volatile storage must be used. Since disk space is limited and used by other programs, magnetic tape is employed.

TABLE 39
DISK FILES FOR ADD/SUBTRACT MODULE TEST PROGRAM, DISK 0

FILE NUMBER	RECORDS/ FILE	WORDS/ RECORD	FUNCTION
1	1	16	PROGRAM PARAMETERS
2	512	13	TEST PATTERNS AND SIMULATOR OUTPUT
3	512	20	ERROR LOG

11.2.4.2 Major Subroutine Functions

Figure 127 shows the major subroutines and disk files accessed by this program. The main program is AST and calls on the three major subroutines in response to instructions by the user. The main program in this case also compiles the data array from the disk files for use by the subroutines. The major subroutine functions are:

1. DATED - Test Pattern Generation
2. ASAT - Auto Test and Error Mask
3. ASMT - Manual Intervention

Figure 128 shows the subroutine structure for the program.

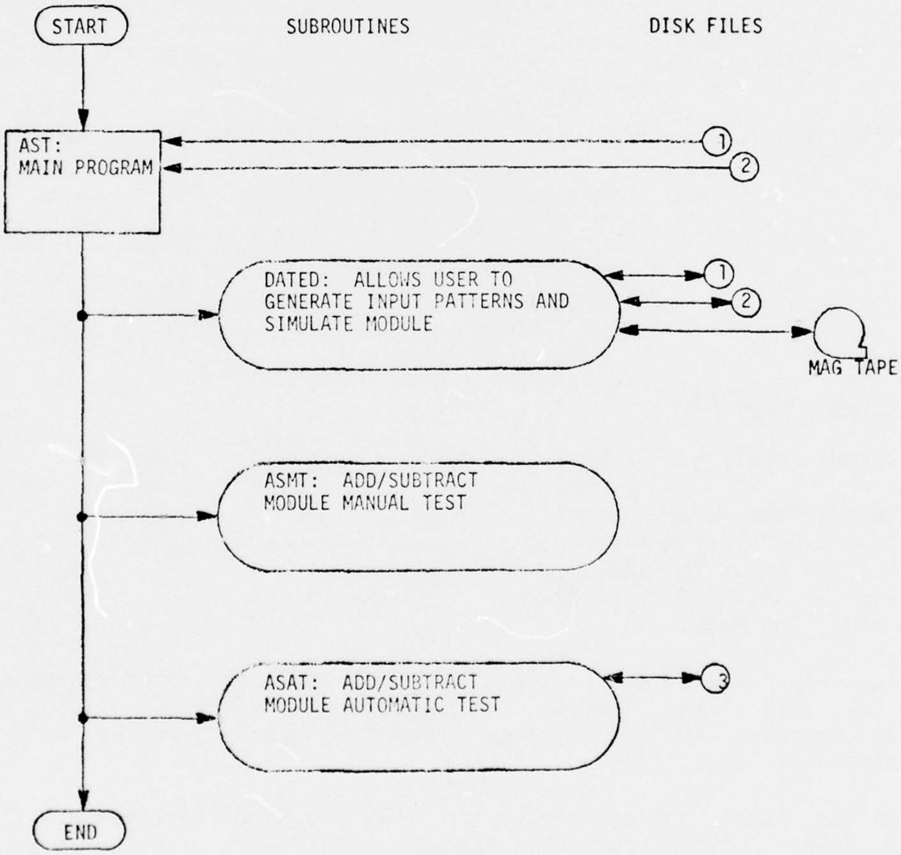


FIGURE 127 MAJOR SUBROUTINE FUNCTIONS AND DISK FILES ACCESSED FOR ADD/ SUBTRACT MODULE SOFTWARE

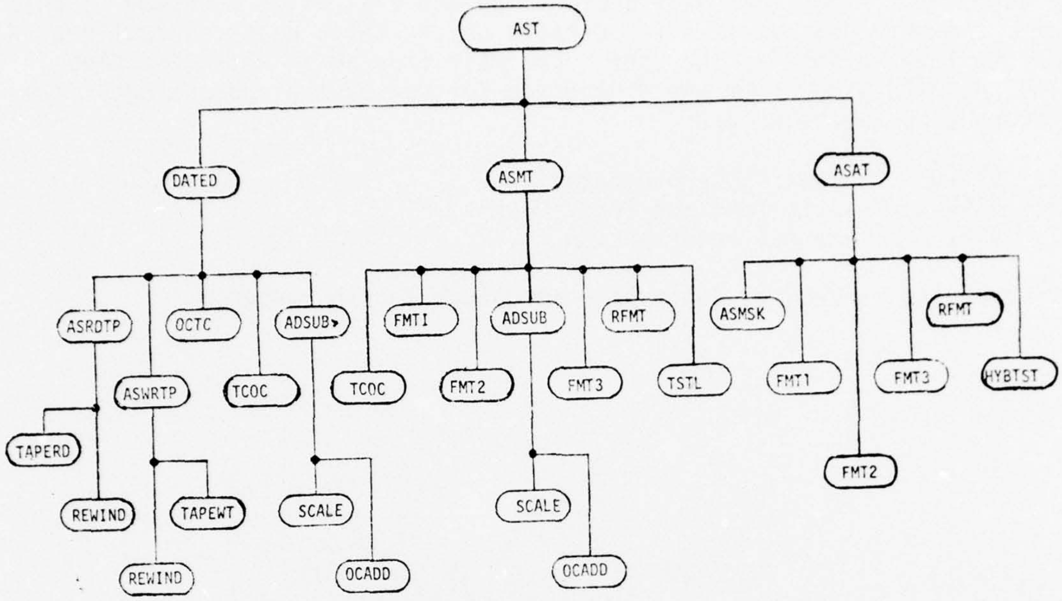


FIGURE 128. SUBROUTINE STRUCTURE FOR ADD/SUBTRACT MODULE TEST PROGRAM

11.2.4.3 Subroutine DATED

This subroutine allows the user to perform the following functions:

1. Zero the Data Array
2. Read the Array from Tape
3. Write the Array to Tape
4. Change a Specified Array Element
5. Simulate the Module

The data array contains an input pattern field and a simulator response field. Any time data is read from or written to magnetic tape both fields are involved. Whenever the "change array element" option is executed, only the input pattern field is modified.

When the "simulator" option is executed, only the simulator response field is modified. When the "zero" option is executed, all fields are cleared as well as the parameter record at the beginning of the tape. Before exiting the program, the data array is written to the disk file.

The simulator subroutine is an assembly language subroutine which performs the adder/subtractor module functions on a bit basis. Data into the module is converted to an 8 bit plus sign ones complement word with 4 bits of floating point exponent. The subroutine is partitioned to perform the same functions as each chip on the actual hardware module. The computer hardware registers are used in as close a way as possible to the hardware to shift, add, subtract and increment exponents. Much of the gate level logic is performed the same way as in the hardware. An example is the exclusive OR function which detects the all one's condition in the hardware.

11.2.4.4 Subroutine ASAT

This program is the automatic test routine. The input pattern field of the data array is sequenced through the adder/subtractor module and the outputs are recorded. The error detection subroutine is also contained in this program. In this subroutine, the adder/subtractor module response to the input patterns is compared to the simulator response field of the data array. The number of errors is recorded and the erroneous responses are written to the disk. The number of errors are indicated to the user and he has the option to print the erroneous responses.

11.2.4.5 Subroutine ASMT

This subroutine is the manual intervention program for the adder/subtractor module test program. The program allows the user to select an input pattern from either the input pattern field of the data array or the keyboard. If the input comes from the keyboard, the pattern is processed by the simulator to form a basis of comparison with the module response. The input pattern is then continuously cycled through the hardware until a signal is given by the user. When this signal is given, the computer reads the module response and displays the simulator response and the module response on the video terminal.

11.2.5 Control Module, Level Translator, and Universal SOS Module Test Programs

These three programs are all subroutine structured user interactive programs which have the same basic features as the other programs. However, this is where the similarity ends. The major functions such as automatic test and manual intervention exist in the body of the main program and not as subroutines. Also no disk files are used.

The test pattern arrays for these modules are so small that they are contained in block data statements either at the beginning of the program or as a separate block data subroutine which is linked to the main program. For this reason, these programs are not re-entrant programs like the previous programs. They must be exited and restarted upon completion of an error mask operation or before recycling through an automatic test. Figures 129 through 131 show the subroutine structure of the three programs.

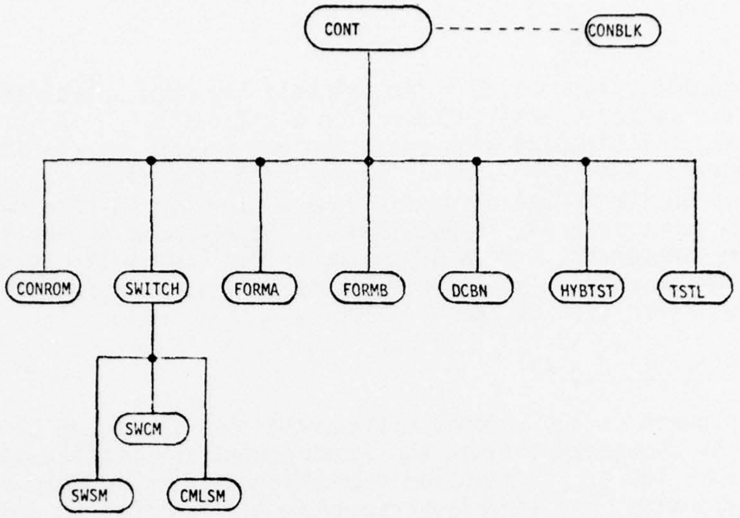


FIGURE 129. SUBROUTINE STRUCTURE FOR CONTROL MODULE TEST PROGRAM

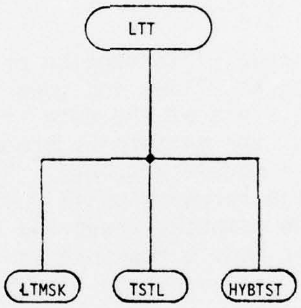


FIGURE 130. SUBROUTINE STRUCTURE FOR LEVEL TRANSLATOR MODULE TEST PROGRAM

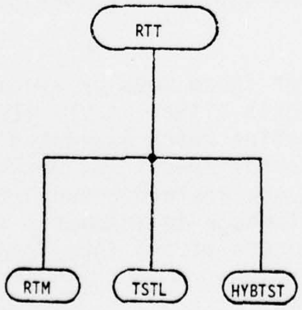


FIGURE 131. SUBROUTINE STRUCTURE FOR UNIVERSAL SOS MODULE TEST PROGRAM

SECTION XII

TEST PROGRAM RESULTS

12.1 MODULE TEST RESULTS

Six module types were tested in the module test bed. The universal TTL modules were not tested since each module represented a unique function and was easily tested in the system because of its low circuit density. The general test procedure was to extensively bench test the first piece of each module type to verify the logic, check the substrate, and determine the speed limitations of the module. When it was determined that there would be no problems with the substrate or logic, the module was released for quantity fabrication. The modules were then tested in the test bed and non-functional modules returned for repairs. The repaired modules were then re-tested and faulty units sent through the cycle again. Failures on a second test were generally due to problems which were masked by other problems on the first test. These problems were usually due to shorts, intermittents, or time dependent failures.

Table 40 shows the results of the first test cycle. These modules represent newly fabricated, untested modules.

TABLE 40
INITIAL TEST RESULTS

MODULE TYPE	NUMBER TESTED	NUMBER FAULTY MODULES	PERCENT FAULTY
FFT Memory	22	12	54.5
Control	18	15	83.3
Retimer-Univ. SOS	13	5	38.5
Level Translator	14	7	50.0
Multiplier	9	4	44.4
Adder/ Subtractor	26	9	34.6

The very high failure rate of the control module was due to a higher susceptability of the bonding wires of the bulk CMOS devices on this module to the ultrasonic cleaning process. It is believed that the high failure rate of the other modules was also primarily due to the ultrasonic cleaning. However, the adder/subtractor modules and the universal SOS-retimer modules were not ultrasonically cleaned. These modules reflect actual bad circuits or substrate trace faults. The FFT memory module had several problems, all of which are reflected in its high initial failure. The first group of memories (10 units) was ultrasonically cleaned and in addition had GUA chips with the instability problem. A second group (12 units) was not ultrasonically cleaned but had GUA chips which were insufficiently screened at SSTC for the instability problem. This group had a slightly lower initial failure rate.

The bad modules were returned to the hybrid lab for repairs and then retested. Table 41 shows the cycle 2 test results.

TABLE 41
CYCLE 2 TEST RESULTS

MODULE TYPE	NUMBER TESTED	NUMBER OF FAILURES	PERCENT FAILURES
FFT Memory	12	5	41.6
Control	15	8	53.3
Level Translator	4	1	25.0
Multiplier	3	0	0
Adder/ Subtractor	9	2	22.2

Three of the FFT memory failures were due to substrate shorts which were not picked up in the first test. The other two were due to bad retimer chips which were installed on the modules as a result of the first test. All of the control modules had bad retimer circuits which were not reintroduced on the initial repair cycle because they were masked by other failures. The two adder/subtractor failures resulted from one shorted trace and one poorly mounted circuit.

Only the adder/subtractor and FFT memory modules were returned for cycle 3 repairs. There were no failures during cycle 3 testing. The control modules, level translator and retimer modules were not repaired because there was a sufficient quantity of these modules to implement the forward and inverse FFT's.

Two multipliers, two adder/subtractors and four FFT memories failed during system test. Three of these failures were hard failures while the other five were time failures which recovered and were thought to be residuals of the instability problem.

Table 42 gives a breakdown of the individual chip failures. Most of the initial chip failures were due to ultrasonic cleaning.

12.2 SYSTEM TEST RESULTS

12.2.1 System Debugging Procedures

Initial operation of the system was at 1 MHz in order to reduce the possibility of timing problems. It was felt that if the control system checked out and the modules were tested that the only problems encountered would be wiring errors. The system test bed was used in the 44 bit probe mode during initial check out.

TABLE 42
CIRCUIT FAILURE DATA (NOT INCLUDING CURRENTLY FAULTY MODULES)

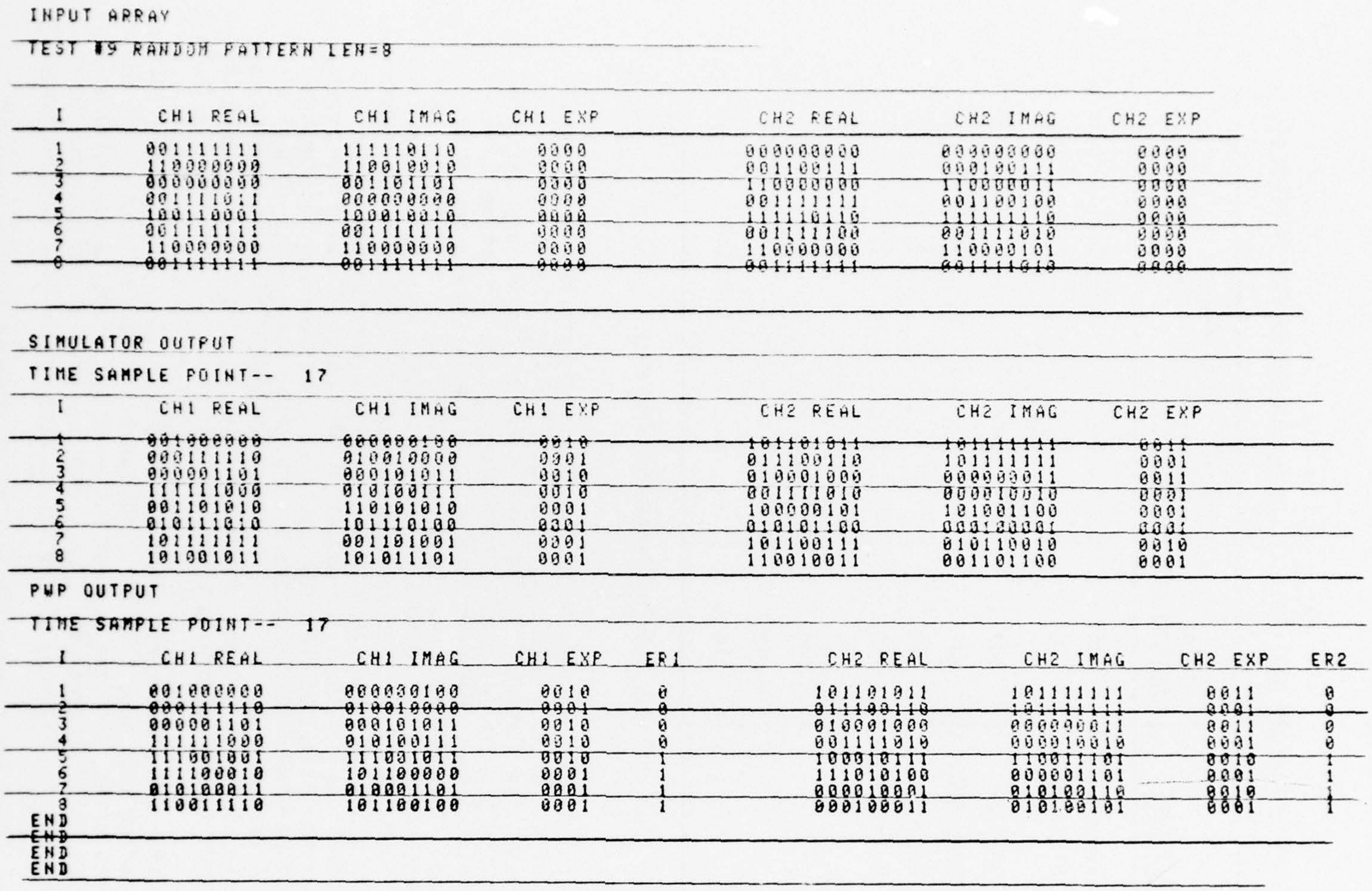
CIRCUIT TYPE	TOTAL ASSEMBLED	FAILURES	
		INITIAL	OPERATING
TCS-016 Scaler	26	2	0
TCS-017 Floating Point	26	2	0
TCS-015 Retimer	197	16*	5
TCS-060-900b GUA	242	6	2
TCS-065 Adder	70	5	0
TCS-057 Multiplier	36	2	1
4019	51	12*	0
75365	136	1	1
LS174	42	1	0
PROMS	14	0	20

* Most Initial Failures Due To Ultrasonic Cleaning

A test pattern was introduced at the FFT input with the 44 bit input probe. The output of each module was then sampled with the 44 bit output probe. The sampling started with the first stage of the FFT and then progressed down the pipeline as modules were plugged into the system. The computer simulated the system response up to the module whose output was being sampled. This response was compared to the sampled response and any erroneous patterns were indicated with an error marker next to the bad response. Figure 132 shows a computer print out of a typical test. The first pattern shows the input to both the simulator and the PWP. The second pattern shows the simulator output at test point 17 along the pipeline. The third pattern shows the PWP output at test point 17 along the pipeline. The columns labeled ER1 and ER2 represent the error markers for channel 1 and channel 2 respectively. A zero indicates a match with the corresponding pattern in the simulator output. A one indicates an error. Notice that the last four words in the example have errors.

If an error was encountered along the line when there was no error on the modules preceding this one, then the probe was moved to the input of the module. If no error occurred, then the problem was an output short, a control problem, or a bad module. If an error did occur, then the problem was an input short or wiring error. In this way, wiring errors were fairly quickly eliminated.

However, after about three quarters of the forward FFT was debugged, control problems began to occur in earlier stages. It became inconvenient to use the output probe to search for these problems because it was relatively



```
INPUT ARRAY
TEST #9 RANDOM PATTERN LEN=8

  I       CH1 REAL      CH1 IMAG    CH1 EXP        CH2 REAL      CH2 IMAG    CH2 EXP

  1      001111111     111110110      0000        000000000     000000000      0000
  2      110000000     110010010      0000        001100111     000100111      0000
  3      000000000     001101101      0000        110000000     110000011      0000
  4      001111011     000000000      0000        001111111     001100100      0000
  5      100110001     100010010      0000        111110110     111111110      0000
  6      001111111     001111111      0000        001111100     001111010      0000
  7      110000000     110000000      0000        110000000     110000101      0000
  8      001111111     001111111      0000        001111111     001111010      0000

SIMULATOR OUTPUT
TIME SAMPLE POINT--   17
  I       CH1 REAL      CH1 IMAG    CH1 EXP        CH2 REAL      CH2 IMAG    CH2 EXP

  1      001000000     000000100      0010        101101011     101111111      0011
  2      000111110     010010000      0001        011100110     101111111      0001
  3      000001101     000101011      0010        010001000     000000011      0011
  4      111111000     010100111      0010        001111010     000010010      0001
  5      001101010     110101010      0001        100000101     101001100      0001
  6      010111010     101110100      0001        010101100     000120001      0001
  7      101111111     001101001      0001        101100111     010110010      0010
  8      101001011     101011101      0001        110010011     001101100      0001
PWP OUTPUT
TIME SAMPLE POINT--   17
  I       CH1 REAL      CH1 IMAG    CH1 EXP   ER1       CH2 REAL      CH2 IMAG    CH2 EXP   ER2

  1      001000000     000030100      0010     0       101101011     101111111      0011     0
  2      000111110     010010000      0001     0       011100110     101111111      0001     0
  3      000001101     000101011      0010     0       010001000     000000011      0011     0
  4      111111000     010100111      0010     0       001111010     000010010      0001     0
  5      111001001     111031011      0010     1       100010111     110011101      0010     1
  6      111100010     101100000      0001     1       111010100     000001101      0001     1
  7      010100011     010001101      0001     1       000010001     010100110      0010     1
  8      110011110     101100100      0001     1       000100011     010100101      0001     1
END
END
END
END
```

FIGURE 132. COMPUTER PRINT OUT OF A TYPICAL TEST

difficult to plug the 44 probe wires into the back plane. In this case, a Hewlett Packard 32 channel logic state analyzer was used to probe the backplane. It was eventually discovered that some of the control PROM data locations had become erroneously programmed. This situation was corrected but occurred again. In addition to control problems, it was discovered that undershoot on the data input lines of the 44 bit input probe caused problems with the TCS-015 retimer chips. These problems were found by probing with oscilloscopes and the logic state analyzer.

After these problems were solved, the forward FFT was extensively exercised at 1 MHz with several different input waveforms. Some waveforms heavily exercised the earlier stages while others exercised the later stages. The FFT length was changed and each configuration was extensively tested.

At this point, the frequency was raised to 2 MHz. No problems were encountered and the frequency was again raised to 4 MHz. At this point, it was discovered that a significant undershoot on the clock lines caused a similar problem with the TCS-015 retimer chips as the earlier problem with the input probe. The undershoot was eliminated by increasing the series damping resistor value on the output of every clock driver. This, however, increased the rise and fall times of the clock edges and introduced skews between clock drivers. It is difficult to tell without further investigation if this contributed anything to the 8 MHz rate which was finally reached. After the clock undershoot was corrected, the rate was again increased. A maximum of 5.2 MHz was reached for the forward FFT. It was determined that this limitation was due to some slow devices in the system. The first two stages of the forward FFT were operated at 8 MHz. This limitation was again due to slow devices.

12.2.2 Special Problems/Solutions

A number of problems which were encountered during system tests will be expanded on here.

During operation, it was found that random bits of the control PROM's were becoming programmed. Three possible causes were identified. First it was found that when the system test bed probe card was inserted into or pulled from the backplane, it was possible for the plug contacts to wipe against exposed contacts leading to the PROM bits. This condition was eliminated by insulating the surfaces and as a further precaution, the probe was not withdrawn or inserted with the power on. Additional bits were becoming programmed after these steps. The only other cause, identified after discussion with Texas Instruments, the PROM manufacturer, was power supply transients. No power transients could be detected and a 7500 μfd capacitor is attached to the V_{CC} bus. A zener diode clamp was placed on the PROM's to further prevent V_{CC} over-voltage. After the zener diodes were added, no further bits became programmed. However, coincident with this last step was the depletion of PROM's with a specific date code. Although a poor batch was suspected, TI claimed that no date code dependent problems had been reported. A final determination between a poor batch and power supply transients has not been made.

During checkout a number of problems were found in addition to the normally expected bad circuits and wiring errors. One of these was an effect observed on the retimers which appeared to lose data about half way through the clock cycle. This was at first attributed to faulty circuits, but further examination revealed that the clock had a much larger undershoot at some points in the system than expected. Substantial tests had been made during the design to determine the proper design of the modules. However, these tests did not use a duplication of the module and backplane hardware and it must be presumed that the model was not sufficiently representative of worst case conditions. The clock undershoot could essentially cause a condition where the data state stored on a chip had to supply chip power as the clock went negative momentarily. If this negative pulse were of sufficient depth and width, the stored "one" state could be lost. Series damping resistors were added to the clock drivers to eliminate this problem. Unfortunately, this also slows the system down by about 10 nsec and the design needs further refinement.

Some circuits were found to fail after being in operation up to several hours. It is not known whether this problem is due to residual hydrogen ion contamination or some other cause. At any rate, until assurance of elimination of this basic problem can be had from fabrication, circuits placed in a system should be dynamically burned in at elevated temperatures to screen for this problem.

The backplane was machine wired with teflon coated wire. It was found that shorts occurred during checkout due to the insulation on one wire pressing against a pin in the case of a 90 degree turn. The teflon insulation would slowly flow out under continued pressure. This is an unusual condition caused by improper tension applied during wirewrap. It would normally not be a problem if point-to-point wiring were used on the backplane.

Clock timing and data undershoot problems were encountered on the test bed interface as the clock rate was increased. The timing problem was solved by using independent laboratory clock generators for each unit although a common clock is a preferred solution. The data undershoot problem which occurred in the remote probe card was eliminated by increasing the decoupling, and adding diode clamps and terminating resistors.

The final problem came to light when the speed capability did not meet expectations. It was initially found that retimers made during a certain period had propagation delays of up to 40 nsec rather than 15 nsec. Further checking into this at SSTC revealed that the conductivity was lower for these runs. Further testing of multiplier circuits also indicated propagation delays up to 35 nsec more than the maximum anticipated. Based on these additional delays of 60 nsec plus 10 nsec due to series clamping resistor on the clocks the maximum operating speed would be 5.9 MHz. The forward FFT has not been exercised without error beyond 5.2 MHz. The speed problem can be avoided by proper wafer screening. SSTC estimates that their yield would decrease by 10-20% if the suspect lower conductivity runs were eliminated.

12.2.3 Performance Measurements

A library of thirty test waveforms was written which included various frequency waveforms, random patterns, DC and impulse patterns. These test patterns were used to exercise the PWP and compare with the simulator responses. The tests were performed at 5 MHz and for all three FFT lengths. In every case the PWP performed exactly as the simulation with the exception of an occasional 1 bit error. *Three of the test patterns and PWP outputs are illustrated here.* The first, Figure 133a and 133b, is a medium frequency alternating square wave. As expected, an impulse occurs near the center of the aperture. This case was run for a 64 point aperture. The second case, Figure 134a and 134b, is a random pattern. This pattern exercises more of the hardware than any of the other patterns. The case was run for a 32 point aperture. There are a few points with a one bit error in this pattern. It is felt that these represent hardware problems and not an algorithm airthmetic error. The third pattern, Figure 135a and 135b is an impulse pattern for a 16 point aperture. The output is a step function and there are no errors in the pattern.

```
INPUT ARRAY
MEDIUM FREQUENCY PATTERN LEN=32

 I      CH1 REAL     CH1 IMAG    CH1 EXP       CH2 REAL     CH2 IMAG    CH2 EXP

 1      001111111    001111111    0000         001111111    001111111    0000
 2      001111111    001111111    0000         001111111    001111111    0000
 3      001111111    001111111    0000         001111111    001111111    0000
 4      001111111    001111111    0000         001111111    001111111    0000
 5      001111111    001111111    0000         001111111    001111111    0000
 6      110000000    110000000    0000         110000000    110000000    0000
 7      110000000    110000000    0000         110000000    110000000    0000
 8      110000000    110000000    0000         110000000    110000000    0000
 9      110000000    110000000    0000         110000000    110000000    0000
10      110000000    110000000    0000         110000000    110000000    0000
11      001111111    001111111    0000         001111111    001111111    0000
12      001111111    001111111    0000         001111111    001111111    0000
13      001111111    001111111    0000         001111111    001111111    0000
14      001111111    001111111    0000         001111111    001111111    0000
15      001111111    001111111    0000         001111111    001111111    0000
16      110000000    110000000    0000         110000000    110000000    0000
17      110000000    110000000    0000         110000000    110000000    0000
18      110000000    110000000    0000         110000000    110000000    0000
19      110000000    110000000    0000         110000000    110000000    0000
20      110000000    110000000    0000         110000000    110000000    0000
21      001111111    001111111    0000         001111111    001111111    0000
22      001111111    001111111    0000         001111111    001111111    0000
23      001111111    001111111    0000         001111111    001111111    0000
24      001111111    001111111    0000         001111111    001111111    0000
25      001111111    001111111    0000         001111111    001111111    0000
26      110000000    110000000    0000         110000000    110000000    0000
27      110000000    110000000    0000         110000000    110000000    0000
28      110000000    110000000    0000         110000000    110000000    0000
29      110000000    110000000    0000         110000000    110000000    0000
30      110000000    110000000    0000         110000000    110000000    0000
31      001111111    001111111    0000         001111111    001111111    0000
32      001111111    001111111    0000         001111111    001111111    0000
```

FIGURE 133a. MEDIUM FREQUENCY ALTERNATING SQUARE WAVE

```
PWP OUTPUT
TIME SAMPLE POINT--  20
   I      CH1 REAL    CH1 IMAG   CH1 EXP  ER1      CH2 REAL    CH2 IMAG   CH2 EXP  ER2

   1      001111110   001111110   0010     0       010111101   010111101   0011     0
   2      100000101   011111110   0001     0       011111110   100000101   0001     0
   3      100101000   111111111   0011     0       110110110   111111110   0010     0
   4      111111110   110110110   0010     0       111111111   100101000   0011     0
   5      010010001   000111100   0011     0       111011000   111101111   0010     0
   6      001100001   100010101   0011     0       000011101   110111001   0011     0
   7      110111001   000011101   0011     0       100010101   001100001   0011     0
   8      111101111   111011000   0010     0       000111100   010010001   0011     0
   9      010101001   001110001   0010     0       111011110   111101010   0010     0
  10      110010011   010100010   0011     0       001010001   110000110   0010     0
  11      110011010   000010011   0011     0       110111110   000001101   0011     0
  12      111111100   111110011   0100     0       000101010   011010110   0101     0
  13      011010110   000101010   0101     0       111110011   111111100   0100     0
  14      000001101   110111110   0011     0       000010011   110011010   0011     0
  15      110000110   001010001   0010     0       010100010   110010011   0011     0
  16      111101010   111011110   0010     0       001110001   010101001   0010     0
  17      000000000   000000000   0000     0       000000000   000000000   0000     0
  18      000000000   000000000   0000     0       000000000   000000000   0000     0
  19      000000000   000000000   0000     0       000000000   000000000   0000     0
  20      000000000   000000000   0000     0       000000000   000000000   0000     0
  21      000000000   000000000   0000     0       000000000   000000000   0000     0
  22      000000000   000000000   0000     0       000000000   000000000   0000     0
  23      000000000   000000000   0000     0       000000000   000000000   0000     0
  24      000000000   000000000   0000     0       000000000   000000000   0000     0
  25      000000000   000000000   0000     0       000000000   000000000   0000     0
  26      000000000   000000000   0000     0       000000000   000000000   0000     0
  27      000000000   000000000   0000     0       000000000   000000000   0000     0
  28      000000000   000000000   0000     0       000000000   000000000   0000     0
  29      000000000   000000000   0000     0       000000000   000000000   0000     0
  30      000000000   000000000   0000     0       000000000   000000000   0000     0
  31      000000000   000000000   0000     0       000000000   000000000   0000     0
  32      000000000   000000000   0000     0       000000000   000000000   0000     0
END
END
END
END
```

FIGURE 133b. MEDIUM FREQUENCY ALTERNATING SQUARE WAVE

```
INPUT ARRAY

TEST #23 RANDOM PATTERN LEN=16

  I    CH1 REAL    CH1 IMAG   CH1 EXP      CH2 REAL    CH2 IMAG   CH2 EXP

  1    110011000   001111011    0000       000101011   010000100    0000
  2    001110011   001000111    0000       001101101   110000010    0000
  3    001110111   000000000    0000       001110100   000000000    0000
  4    001101111   001100110    0000       001111111   001111111    0000
  5    001110001   111110101    0000       001100111   110000000    0000
  6    000000000   000000001    0000       000000111   000000011    0000
  7    000000010   000001010    0000       111111100   111110110    0000
  8    000000000   000010111    0000       111010010   001100011    0000
  9    001101011   110011001    0000       110011011   110011100    0000
 10    001100010   110101111    0000       001000011   000100010    0000
 11    001100100   000111000    0000       001001101   001100011    0000
 12    111010010   111101001    0000       111011101   000011010    0000
 13    001100101   111100110    0000       111101110   000011101    0000
 14    001100011   010000010    0000       001111000   001110111    0000
 15    000000000   000000000    0000       000000000   000000000    0000
 16    000110111   001001110    0000       001100111   000110001    0000
```

FIGURE 134a. RANDOM PATTERN

```
PWP OUTPUT

TIME SAMPLE POINT--  19

  I    CH1 REAL    CH1 IMAG   CH1 EXP  ER1      CH2 REAL    CH2 IMAG   CH2 EXP  ER2

  1    010111111   001010111   0011     0       111000111   101110111   0010     0
  2    110000101   101101111   0010     0       000101010   000011100   0010     0
  3    110100011   101100111   0010     0       111110100   001010010   0001     0
  4    100101100   010101010   0010     0       111010110   001111110   0010     0
  5    000100110   001001011   0010     0       101010001   000001011   0010     0
  6    000011110   010000110   0011     0       101111101   111011110   0010     0
  7    110001011   001100010   0010     0       001001110   011100100   0010     0
  8    111000001   001001010   0010     0       010101100   010111000   0010     0
  9    010101001   000011011   0001     0       000100110   010100010   0001     0
 10    110100110   011001001   0000     1       011000111   001010000   0001     1
 11    111010101   000011110   0000     0       010010010   010100101   0001     0
 12    111000011   010000100   0001     0       010101000   010110010   0001     0
 13    101101011   110110111   0001     0       010010000   010000110   0001     0
 14    111101010   001001100   0001     1       001000101   110010001   0001     1
 15    000010101   100011111   0001     1       010000010   110111100   0010     0
 16    011111001   101111100   0001     0       111111110   101011100   0010     0
END
END
END
END
```

FIGURE 134b. RANDOM PATTERN

```
INPUT ARRAY
TEST #20 COMPLEX IMP ALL CHAN LEN=8 LOC =1

  I        CH1 REAL       CH1 IMAG     CH1 EXP         CH2 REAL       CH2 IMAG     CH2 EXP

  1        011111111      011111111     0000           011111111      011111111     0000
  2        000000000      000000000     0000           000000000      000000000     0000
  3        000000000      000000000     0000           000000000      000000000     0000
  4        000000000      000000000     0000           000000000      000000000     0000
  5        000000000      000000000     0000           000000000      C00000000     0000
  6        000000000      000000000     0000           000000000      000000000     0000
  7        000000000      000000000     0000           000000000      000000000     0000
  8        000000000      000000000     0000           000000000      000000000     0000
```

FIGURE 135a. IMPULSE PATTERN

```
PWP OUTPUT
TIME SAMPLE POINT--   17
 I        CH1 REAL       CH1 IMAG     CH1 EXP   ER1        CH2 REAL       CH2 IMAG     CH2 EXP   ER2

 1        011111110      011111110     0001     0          011111110      011111110     0001     0
 2        011111110      011111110     0001     0          011111110      011111110     0001     0
 3        011111110      011111110     0001     0          011111110      011111110     0001     0
 4        011111110      011111110     0001     0          011111110      011111110     0001     0
 5        000000001      000000001     0000     0          000000001      000000001     0000     0
 6        000000001      000000001     0000     0          000000001      000000001     0000     0
 7        000000001      000000001     0000     0          000000001      000000001     0000     0
 8        000000001      000000001     0000     0          000000001      000000001     0000     0
END
END
END
END
```

FIGURE 135b. IMPULSE PATTERN

Preceding Page Blank - NOT FILMed

REFERENCES

1. R. P. Perry, R. J. Smith, L. W. Martinson, "Pulse Compression Techniques", AFAL-TR-73-143, June 1973.

2. L. W. Martinson, R. P. Perry, R. J. Smith, et. al., "Programmable Waveform Generator, Phase I", AFAL-TR-74-120, June 1974.

3. R. P. Perry, H. W. Kaiser, "Digital Step Transform Approach to Airborne Radar Processing", NAECON '73 Convention Record, April 1973.

4. R. O. Harger, Synthetic Aperture Radar Systems, Academic Press, New York, 1970.

5. W. M. Brown, "Synthetic Aperture Radar", IEEE Trans., Aerospace and Electronic Systems, Vol. AES-3, March 1967.

6. Camel Signal Processing Study, RADC-TR-70-266, October 1970 (Secret).

7. L. W. Martinson, R. J. Smith, "Digital Matched Filtering With Pipelined Floating Point Fast Fourier Transforms", IEEE Transactions on Acoustics, Speech and Signal Processing, ASSP-23, Number 2, April 1975.

8. D. S. Woo, S. G. Policastro, H. Borkan, "Manufacturing Methods for Silicon Devices on Insulating Substrates", Technical Report AFML-IR-516-3 (III, IV), April 30, 1974 and July 30, 1974.

9. G. E. Moore, "What Level of LSI is Best For You?", Electronics 43:126, February 16, 1970.

10. Skolnik, M., Radar Handbook. McGraw-Hill, 1970, p. 20-30.

☆ U. S. GOVERNMENT PRINTING OFFICE: 1977 — 757-001/561